

# Package ‘remverse’

July 18, 2026

**Title** Comprehensive Tools for Relational Event History Data

**Version** 0.2.0

**Description** A unified interface for relational event history analysis.

The package re-exports key functions from 'remify', 'remstats', and 'remstimate' to support a streamlined workflow from data processing to model estimation and diagnosis.

**License** MIT + file LICENSE

**BugReports** <https://github.com/TilburgNetworkGroup/remverse/issues>

**Encoding** UTF-8

**RoxygenNote** 7.3.3

**Imports** remify (>= 4.1.0), remstats (>= 4.1.0), remstimate (>= 3.1.0)

**Depends** R (>= 4.0), remdata (>= 0.2.1)

**Suggests** tinytest, knitr, rmarkdown, survival, coxme, lme4, glmmTMB,  
glmnet, flexmix, shrinkem (>= 0.4.0)

**LazyData** true

**VignetteBuilder** knitr

**LazyDataCompression** gzip

**NeedsCompilation** no

**Author** Joris Mulder [aut, cre],  
Giuseppe Arena [aut],  
Marlyne Meijerink-Bosman [aut],  
Rumana Lakdawala [aut],  
Roger Leenders [aut],  
Fabio Generoso Vieira [aut],  
Mahdi Shafiee Kamalabad [aut],  
Diana Karimova [ctb]

**Maintainer** Joris Mulder <j.mulder3@tilburguniversity.edu>

**Repository** CRAN

**Date/Publication** 2026-07-18 18:40:02 UTC

## Contents

|                                     |    |
|-------------------------------------|----|
| activeDegreeDyad . . . . .          | 4  |
| activeDegreeMax . . . . .           | 4  |
| activeDegreeMin . . . . .           | 5  |
| activeIndegreeReceiver . . . . .    | 6  |
| activeOutdegreeSender . . . . .     | 6  |
| activeReciprocalTie . . . . .       | 7  |
| activeSharedPartners . . . . .      | 8  |
| activeSharedPartners_isp . . . . .  | 8  |
| activeSharedPartners_itp . . . . .  | 9  |
| activeSharedPartners_osp . . . . .  | 10 |
| activeSharedPartners_otp . . . . .  | 10 |
| activeTie . . . . .                 | 11 |
| activeTotaldegreeDyad . . . . .     | 12 |
| activeTotaldegreeReceiver . . . . . | 12 |
| activeTotaldegreeSender . . . . .   | 13 |
| actor_effects . . . . .             | 14 |
| AICC . . . . .                      | 14 |
| aomstats . . . . .                  | 15 |
| ao_data . . . . .                   | 16 |
| average . . . . .                   | 16 |
| bic_table . . . . .                 | 17 |
| bind_remstats . . . . .             | 17 |
| degreeDiff . . . . .                | 18 |
| degreeMax . . . . .                 | 18 |
| degreeMin . . . . .                 | 19 |
| diagnostics . . . . .               | 19 |
| difference . . . . .                | 20 |
| dlcrem . . . . .                    | 21 |
| dyad . . . . .                      | 21 |
| edgelist_duration . . . . .         | 22 |
| event . . . . .                     | 22 |
| FEtype . . . . .                    | 23 |
| frailty_rem . . . . .               | 23 |
| history_durem . . . . .             | 23 |
| indegreeReceiver . . . . .          | 24 |
| indegreeSender . . . . .            | 25 |
| inertia . . . . .                   | 25 |
| info3 . . . . .                     | 26 |
| is.remify_durem . . . . .           | 26 |
| is.remstats_durem . . . . .         | 27 |
| isp . . . . .                       | 27 |
| itp . . . . .                       | 28 |
| maximum . . . . .                   | 28 |
| minimum . . . . .                   | 29 |
| osp . . . . .                       | 29 |
| otp . . . . .                       | 30 |

|                                  |    |
|----------------------------------|----|
| outdegreeReceiver . . . . .      | 31 |
| outdegreeSender . . . . .        | 31 |
| psABA . . . . .                  | 32 |
| psABAB . . . . .                 | 32 |
| psABAY . . . . .                 | 33 |
| psABB . . . . .                  | 33 |
| psABBA . . . . .                 | 33 |
| psABBY . . . . .                 | 34 |
| psABX . . . . .                  | 34 |
| psABXA . . . . .                 | 35 |
| psABXB . . . . .                 | 35 |
| psABXY . . . . .                 | 36 |
| randomREH . . . . .              | 36 |
| randomREH3 . . . . .             | 36 |
| receive . . . . .                | 37 |
| recencyContinue . . . . .        | 38 |
| recencyReceiveReceiver . . . . . | 38 |
| recencyReceiveSender . . . . .   | 39 |
| recencySendReceiver . . . . .    | 39 |
| recencySendSender . . . . .      | 40 |
| reciprocity . . . . .            | 40 |
| remfrailty . . . . .             | 41 |
| remify . . . . .                 | 41 |
| remixture . . . . .              | 45 |
| rempenalty . . . . .             | 45 |
| remstats . . . . .               | 46 |
| remstimate . . . . .             | 48 |
| remtribute . . . . .             | 49 |
| remwindow . . . . .              | 50 |
| rrankReceive . . . . .           | 51 |
| rrankSend . . . . .              | 52 |
| same . . . . .                   | 52 |
| select_stats . . . . .           | 53 |
| send . . . . .                   | 53 |
| sp . . . . .                     | 54 |
| spUnique . . . . .               | 54 |
| stack_stats . . . . .            | 55 |
| tie . . . . .                    | 55 |
| tie_data . . . . .               | 56 |
| tie_effects . . . . .            | 56 |
| tomstats . . . . .               | 57 |
| totaldegreeDyad . . . . .        | 58 |
| totaldegreeReceiver . . . . .    | 58 |
| totaldegreeSender . . . . .      | 59 |
| userStat . . . . .               | 60 |
| WAIC . . . . .                   | 60 |

---

|                  |                         |
|------------------|-------------------------|
| activeDegreeDyad | <i>activeDegreeDyad</i> |
|------------------|-------------------------|

---

### Description

See [activeDegreeDyad](#).

### Usage

```
activeDegreeDyad(scaling = c("none", "std"), consider_type = "ignore")
```

### Arguments

|               |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    |
|---------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| scaling       | Scaling applied to the raw statistic before returning:<br>"none" Raw counts (default).<br>"std" Per-time-point standardisation: $(x - \bar{x})/\text{sd}(x)$ , computed over the $D$ dyads in the fixed risk set.                                                                                                                                                                                                                                                                                                                                                                                  |
| consider_type | Character (or logical). How event types are handled:<br>"ignore" <b>or</b> FALSE Aggregate over all event types (default). Counts all currently active events regardless of type.<br>"separate" <b>or</b> TRUE Compute one statistic per event type. Only active events of that type contribute. Output effect names are suffixed with the type label, e.g. activeOutdegreeSender.X.start.<br>"interact" Compute one statistic per (past-event type x dyad type) combination ( $C^2$ slices). Requires extend_riskset_by_type = TRUE in the remify call; otherwise silently coerced to "separate". |

---

|                 |                        |
|-----------------|------------------------|
| activeDegreeMax | <i>activeDegreeMax</i> |
|-----------------|------------------------|

---

### Description

See [activeDegreeMax](#).

### Usage

```
activeDegreeMax(scaling = c("none", "std"), consider_type = "ignore")
```

**Arguments**

|               |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    |
|---------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| scaling       | Scaling applied to the raw statistic before returning:<br>"none" Raw counts (default).<br>"std" Per-time-point standardisation: $(x - \bar{x})/\text{sd}(x)$ , computed over the $D$ dyads in the fixed risk set.                                                                                                                                                                                                                                                                                                                                                                                  |
| consider_type | Character (or logical). How event types are handled:<br>"ignore" <b>or</b> FALSE Aggregate over all event types (default). Counts all currently active events regardless of type.<br>"separate" <b>or</b> TRUE Compute one statistic per event type. Only active events of that type contribute. Output effect names are suffixed with the type label, e.g. activeOutdegreeSender.X.start.<br>"interact" Compute one statistic per (past-event type x dyad type) combination ( $C^2$ slices). Requires extend_riskset_by_type = TRUE in the remify call; otherwise silently coerced to "separate". |

---

|                 |                        |
|-----------------|------------------------|
| activeDegreeMin | <i>activeDegreeMin</i> |
|-----------------|------------------------|

---

**Description**

See [activeDegreeMin](#).

**Usage**

```
activeDegreeMin(scaling = c("none", "std"), consider_type = "ignore")
```

**Arguments**

|               |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    |
|---------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| scaling       | Scaling applied to the raw statistic before returning:<br>"none" Raw counts (default).<br>"std" Per-time-point standardisation: $(x - \bar{x})/\text{sd}(x)$ , computed over the $D$ dyads in the fixed risk set.                                                                                                                                                                                                                                                                                                                                                                                  |
| consider_type | Character (or logical). How event types are handled:<br>"ignore" <b>or</b> FALSE Aggregate over all event types (default). Counts all currently active events regardless of type.<br>"separate" <b>or</b> TRUE Compute one statistic per event type. Only active events of that type contribute. Output effect names are suffixed with the type label, e.g. activeOutdegreeSender.X.start.<br>"interact" Compute one statistic per (past-event type x dyad type) combination ( $C^2$ slices). Requires extend_riskset_by_type = TRUE in the remify call; otherwise silently coerced to "separate". |

---

activeIndegreeReceiver

*activeIndegreeReceiver*


---

### Description

See [activeIndegreeReceiver](#).

### Usage

```
activeIndegreeReceiver(scaling = c("none", "std"), consider_type = "ignore")
```

### Arguments

|               |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    |
|---------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| scaling       | Scaling applied to the raw statistic before returning:<br>"none" Raw counts (default).<br>"std" Per-time-point standardisation: $(x - \bar{x})/\text{sd}(x)$ , computed over the $D$ dyads in the fixed risk set.                                                                                                                                                                                                                                                                                                                                                                                  |
| consider_type | Character (or logical). How event types are handled:<br>"ignore" <b>or</b> FALSE Aggregate over all event types (default). Counts all currently active events regardless of type.<br>"separate" <b>or</b> TRUE Compute one statistic per event type. Only active events of that type contribute. Output effect names are suffixed with the type label, e.g. activeOutdegreeSender.X.start.<br>"interact" Compute one statistic per (past-event type x dyad type) combination ( $C^2$ slices). Requires extend_riskset_by_type = TRUE in the remify call; otherwise silently coerced to "separate". |

---

activeOutdegreeSender *activeOutdegreeSender*


---

### Description

See [activeOutdegreeSender](#).

### Usage

```
activeOutdegreeSender(scaling = c("none", "std"), consider_type = "ignore")
```

**Arguments**

|               |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    |
|---------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| scaling       | Scaling applied to the raw statistic before returning:<br>"none" Raw counts (default).<br>"std" Per-time-point standardisation: $(x - \bar{x})/\text{sd}(x)$ , computed over the $D$ dyads in the fixed risk set.                                                                                                                                                                                                                                                                                                                                                                                  |
| consider_type | Character (or logical). How event types are handled:<br>"ignore" <b>or</b> FALSE Aggregate over all event types (default). Counts all currently active events regardless of type.<br>"separate" <b>or</b> TRUE Compute one statistic per event type. Only active events of that type contribute. Output effect names are suffixed with the type label, e.g. activeOutdegreeSender.X.start.<br>"interact" Compute one statistic per (past-event type x dyad type) combination ( $C^2$ slices). Requires extend_riskset_by_type = TRUE in the remify call; otherwise silently coerced to "separate". |

---

activeReciprocalTie     *activeReciprocalTie*

---

**Description**

See [activeReciprocalTie](#).

**Usage**

```
activeReciprocalTie(scaling = c("none", "std"), consider_type = "ignore")
```

**Arguments**

|               |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    |
|---------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| scaling       | Scaling applied to the raw statistic before returning:<br>"none" Raw counts (default).<br>"std" Per-time-point standardisation: $(x - \bar{x})/\text{sd}(x)$ , computed over the $D$ dyads in the fixed risk set.                                                                                                                                                                                                                                                                                                                                                                                  |
| consider_type | Character (or logical). How event types are handled:<br>"ignore" <b>or</b> FALSE Aggregate over all event types (default). Counts all currently active events regardless of type.<br>"separate" <b>or</b> TRUE Compute one statistic per event type. Only active events of that type contribute. Output effect names are suffixed with the type label, e.g. activeOutdegreeSender.X.start.<br>"interact" Compute one statistic per (past-event type x dyad type) combination ( $C^2$ slices). Requires extend_riskset_by_type = TRUE in the remify call; otherwise silently coerced to "separate". |

---

activeSharedPartners    *activeSharedPartners*

---

### Description

See [activeSharedPartners](#).

### Usage

```
activeSharedPartners(scaling = c("none", "std"), consider_type = "ignore")
```

### Arguments

|               |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    |
|---------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| scaling       | Scaling applied to the raw statistic before returning:<br>"none" Raw counts (default).<br>"std" Per-time-point standardisation: $(x - \bar{x})/\text{sd}(x)$ , computed over the $D$ dyads in the fixed risk set.                                                                                                                                                                                                                                                                                                                                                                                  |
| consider_type | Character (or logical). How event types are handled:<br>"ignore" <b>or</b> FALSE Aggregate over all event types (default). Counts all currently active events regardless of type.<br>"separate" <b>or</b> TRUE Compute one statistic per event type. Only active events of that type contribute. Output effect names are suffixed with the type label, e.g. activeOutdegreeSender.X.start.<br>"interact" Compute one statistic per (past-event type x dyad type) combination ( $C^2$ slices). Requires extend_riskset_by_type = TRUE in the remify call; otherwise silently coerced to "separate". |

---

activeSharedPartners\_isp  
                           *activeSharedPartners\_isp*

---

### Description

See [activeSharedPartners\\_isp](#).

### Usage

```
activeSharedPartners_isp(scaling = c("none", "std"), consider_type = "ignore")
```

**Arguments**

|               |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    |
|---------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| scaling       | Scaling applied to the raw statistic before returning:<br>"none" Raw counts (default).<br>"std" Per-time-point standardisation: $(x - \bar{x})/\text{sd}(x)$ , computed over the $D$ dyads in the fixed risk set.                                                                                                                                                                                                                                                                                                                                                                                  |
| consider_type | Character (or logical). How event types are handled:<br>"ignore" <b>or</b> FALSE Aggregate over all event types (default). Counts all currently active events regardless of type.<br>"separate" <b>or</b> TRUE Compute one statistic per event type. Only active events of that type contribute. Output effect names are suffixed with the type label, e.g. activeOutdegreeSender.X.start.<br>"interact" Compute one statistic per (past-event type x dyad type) combination ( $C^2$ slices). Requires extend_riskset_by_type = TRUE in the remify call; otherwise silently coerced to "separate". |

---

activeSharedPartners\_itp

*activeSharedPartners\_itp*


---

**Description**

See [activeSharedPartners\\_itp](#).

**Usage**

```
activeSharedPartners_itp(scaling = c("none", "std"), consider_type = "ignore")
```

**Arguments**

|               |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    |
|---------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| scaling       | Scaling applied to the raw statistic before returning:<br>"none" Raw counts (default).<br>"std" Per-time-point standardisation: $(x - \bar{x})/\text{sd}(x)$ , computed over the $D$ dyads in the fixed risk set.                                                                                                                                                                                                                                                                                                                                                                                  |
| consider_type | Character (or logical). How event types are handled:<br>"ignore" <b>or</b> FALSE Aggregate over all event types (default). Counts all currently active events regardless of type.<br>"separate" <b>or</b> TRUE Compute one statistic per event type. Only active events of that type contribute. Output effect names are suffixed with the type label, e.g. activeOutdegreeSender.X.start.<br>"interact" Compute one statistic per (past-event type x dyad type) combination ( $C^2$ slices). Requires extend_riskset_by_type = TRUE in the remify call; otherwise silently coerced to "separate". |

---

activeSharedPartners\_osp  
*activeSharedPartners\_osp*

---

### Description

See [activeSharedPartners\\_osp](#).

### Usage

```
activeSharedPartners_osp(scaling = c("none", "std"), consider_type = "ignore")
```

### Arguments

|               |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    |
|---------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| scaling       | Scaling applied to the raw statistic before returning:<br>"none" Raw counts (default).<br>"std" Per-time-point standardisation: $(x - \bar{x})/\text{sd}(x)$ , computed over the $D$ dyads in the fixed risk set.                                                                                                                                                                                                                                                                                                                                                                                  |
| consider_type | Character (or logical). How event types are handled:<br>"ignore" <b>or</b> FALSE Aggregate over all event types (default). Counts all currently active events regardless of type.<br>"separate" <b>or</b> TRUE Compute one statistic per event type. Only active events of that type contribute. Output effect names are suffixed with the type label, e.g. activeOutdegreeSender.X.start.<br>"interact" Compute one statistic per (past-event type x dyad type) combination ( $C^2$ slices). Requires extend_riskset_by_type = TRUE in the remify call; otherwise silently coerced to "separate". |

---

activeSharedPartners\_otp  
*activeSharedPartners\_otp*

---

### Description

See [activeSharedPartners\\_otp](#).

### Usage

```
activeSharedPartners_otp(scaling = c("none", "std"), consider_type = "ignore")
```

**Arguments**

|               |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    |
|---------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| scaling       | Scaling applied to the raw statistic before returning:<br>"none" Raw counts (default).<br>"std" Per-time-point standardisation: $(x - \bar{x})/\text{sd}(x)$ , computed over the $D$ dyads in the fixed risk set.                                                                                                                                                                                                                                                                                                                                                                                  |
| consider_type | Character (or logical). How event types are handled:<br>"ignore" <b>or</b> FALSE Aggregate over all event types (default). Counts all currently active events regardless of type.<br>"separate" <b>or</b> TRUE Compute one statistic per event type. Only active events of that type contribute. Output effect names are suffixed with the type label, e.g. activeOutdegreeSender.X.start.<br>"interact" Compute one statistic per (past-event type x dyad type) combination ( $C^2$ slices). Requires extend_riskset_by_type = TRUE in the remify call; otherwise silently coerced to "separate". |

---

|           |                  |
|-----------|------------------|
| activeTie | <i>activeTie</i> |
|-----------|------------------|

---

**Description**

See [activeTie](#).

**Usage**

```
activeTie(scaling = c("none", "std"), consider_type = "ignore")
```

**Arguments**

|               |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    |
|---------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| scaling       | Scaling applied to the raw statistic before returning:<br>"none" Raw counts (default).<br>"std" Per-time-point standardisation: $(x - \bar{x})/\text{sd}(x)$ , computed over the $D$ dyads in the fixed risk set.                                                                                                                                                                                                                                                                                                                                                                                  |
| consider_type | Character (or logical). How event types are handled:<br>"ignore" <b>or</b> FALSE Aggregate over all event types (default). Counts all currently active events regardless of type.<br>"separate" <b>or</b> TRUE Compute one statistic per event type. Only active events of that type contribute. Output effect names are suffixed with the type label, e.g. activeOutdegreeSender.X.start.<br>"interact" Compute one statistic per (past-event type x dyad type) combination ( $C^2$ slices). Requires extend_riskset_by_type = TRUE in the remify call; otherwise silently coerced to "separate". |

---

activeTotaldegreeDyad    *activeTotaldegreeDyad*

---

### Description

See [activeTotaldegreeDyad](#).

### Usage

```
activeTotaldegreeDyad(scaling = c("none", "std"), consider_type = "ignore")
```

### Arguments

|               |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    |
|---------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| scaling       | Scaling applied to the raw statistic before returning:<br>"none" Raw counts (default).<br>"std" Per-time-point standardisation: $(x - \bar{x})/\text{sd}(x)$ , computed over the $D$ dyads in the fixed risk set.                                                                                                                                                                                                                                                                                                                                                                                  |
| consider_type | Character (or logical). How event types are handled:<br>"ignore" <b>or</b> FALSE Aggregate over all event types (default). Counts all currently active events regardless of type.<br>"separate" <b>or</b> TRUE Compute one statistic per event type. Only active events of that type contribute. Output effect names are suffixed with the type label, e.g. activeOutdegreeSender.X.start.<br>"interact" Compute one statistic per (past-event type x dyad type) combination ( $C^2$ slices). Requires extend_riskset_by_type = TRUE in the remify call; otherwise silently coerced to "separate". |

---

activeTotaldegreeReceiver  
                                   *activeTotaldegreeReceiver*

---

### Description

See [activeTotaldegreeReceiver](#).

### Usage

```
activeTotaldegreeReceiver(scaling = c("none", "std"), consider_type = "ignore")
```

**Arguments**

|               |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    |
|---------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| scaling       | Scaling applied to the raw statistic before returning:<br>"none" Raw counts (default).<br>"std" Per-time-point standardisation: $(x - \bar{x})/\text{sd}(x)$ , computed over the $D$ dyads in the fixed risk set.                                                                                                                                                                                                                                                                                                                                                                                  |
| consider_type | Character (or logical). How event types are handled:<br>"ignore" <b>or</b> FALSE Aggregate over all event types (default). Counts all currently active events regardless of type.<br>"separate" <b>or</b> TRUE Compute one statistic per event type. Only active events of that type contribute. Output effect names are suffixed with the type label, e.g. activeOutdegreeSender.X.start.<br>"interact" Compute one statistic per (past-event type x dyad type) combination ( $C^2$ slices). Requires extend_riskset_by_type = TRUE in the remify call; otherwise silently coerced to "separate". |

---

activeTotaldegreeSender

*activeTotaldegreeSender*


---

**Description**

See [activeTotaldegreeSender](#).

**Usage**

```
activeTotaldegreeSender(scaling = c("none", "std"), consider_type = "ignore")
```

**Arguments**

|               |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    |
|---------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| scaling       | Scaling applied to the raw statistic before returning:<br>"none" Raw counts (default).<br>"std" Per-time-point standardisation: $(x - \bar{x})/\text{sd}(x)$ , computed over the $D$ dyads in the fixed risk set.                                                                                                                                                                                                                                                                                                                                                                                  |
| consider_type | Character (or logical). How event types are handled:<br>"ignore" <b>or</b> FALSE Aggregate over all event types (default). Counts all currently active events regardless of type.<br>"separate" <b>or</b> TRUE Compute one statistic per event type. Only active events of that type contribute. Output effect names are suffixed with the type label, e.g. activeOutdegreeSender.X.start.<br>"interact" Compute one statistic per (past-event type x dyad type) combination ( $C^2$ slices). Requires extend_riskset_by_type = TRUE in the remify call; otherwise silently coerced to "separate". |

---

|               |                      |
|---------------|----------------------|
| actor_effects | <i>actor_effects</i> |
|---------------|----------------------|

---

**Description**

See [actor\\_effects](#).

**Usage**

actor\_effects(step = NULL)

**Arguments**

step                      outputs all statistics in the sender activity step (if ‘step = sender’) or receiver choice step (if ‘step = receiver’).

---

|      |             |
|------|-------------|
| AICC | <i>AICC</i> |
|------|-------------|

---

**Description**

See [AICC](#).

**Usage**

AICC(object, ...)

**Arguments**

object                    is a remstimate object.  
...                        further arguments to be passed to the ‘AICC’ method.

aomstats

*aomstats***Description**

See [aomstats](#).

**Usage**

```
aomstats(
  reh,
  sender_effects = NULL,
  receiver_effects = NULL,
  memory = c("full", "window", "decay", "interval"),
  memory_value = NA,
  first = 2,
  last = Inf,
  display_progress = FALSE,
  attr_actors = NULL,
  attr_dyads = NULL
)
```

**Arguments**

|                               |                                                                                                                                                                                                                                                                                           |
|-------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <code>reh</code>              | an object of class " <a href="#">remify</a> " characterizing the relational event history. May also be a <code>remify_durem</code> object for duration relational event models.                                                                                                           |
| <code>sender_effects</code>   | an object of class " <a href="#">formula</a> " (or one that can be coerced to that class): a symbolic description of the effects in the sender activity rate step of the actor-oriented model for which statistics are computed, see 'Details'                                            |
| <code>receiver_effects</code> | an object of class " <a href="#">formula</a> " (or one that can be coerced to that class): a symbolic description of the effects in the receiver choice step of model for which statistics are computed, see 'Details'                                                                    |
| <code>memory</code>           | The memory to be used. See 'Details'.                                                                                                                                                                                                                                                     |
| <code>memory_value</code>     | Numeric value indicating the memory parameter. Default is NA, which is only valid for <code>memory = "full"</code> (no memory parameter required). See 'Details'.                                                                                                                         |
| <code>first</code>            | an optional integer value, specifying the index of the first unique time point event in the relational event history for which statistics must be computed (see 'Details'). Default is 2: the first event has no history and is used only to initialize statistics, not to fit the model. |
| <code>last</code>             | an optional integer value, specifying the index of the last unique time point in the relational event history for which statistics must be computed (see 'Details')                                                                                                                       |
| <code>display_progress</code> | should a progress bar for the computation of the endogenous statistics be shown (TRUE) or not (FALSE)?                                                                                                                                                                                    |

|             |                                                                                                                                         |
|-------------|-----------------------------------------------------------------------------------------------------------------------------------------|
| attr_actors | optionally, an object of class " <a href="#">data.frame</a> " that contains exogenous attributes for actors (see Details).              |
| attr_dyads  | optionally, an object of class <code>data.frame</code> or <code>matrix</code> containing attribute information for dyads (see Details). |
| <hr/>       |                                                                                                                                         |
| ao_data     | <i>ao_data</i>                                                                                                                          |
| <hr/>       |                                                                                                                                         |

**Description**

See [ao\\_data](#).

|         |                |
|---------|----------------|
| average | <i>average</i> |
|---------|----------------|

**Description**

See [average](#).

**Usage**

```
average(variable, attr_actors = NULL, scaling = c("none", "std"), attr_data)
```

**Arguments**

|             |                                                                                                                                                                          |
|-------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| variable    | string with the name of the column in the <code>attr_actors</code> object for which the statistic has to be computed.                                                    |
| attr_actors | optionally, an object of class <a href="#">data.frame</a> that contains the attribute, see 'Details.'                                                                    |
| scaling     | the method for scaling the statistic. Default is to not scale the statistic. Alternatively, standardization of the statistic per time point can be requested with "std". |
| attr_data   | Deprecated argument. Please use 'attr_actors' instead.                                                                                                                   |

---

|           |                  |
|-----------|------------------|
| bic_table | <i>bic_table</i> |
|-----------|------------------|

---

**Description**

See [bic\\_table](#).

**Usage**

bic\_table(x, ...)

**Arguments**

- |     |                                                                                             |
|-----|---------------------------------------------------------------------------------------------|
| x   | A remstimate_mixrem_list returned when k is a vector in remstimate(..., method = "MIXREM"). |
| ... | Unused.                                                                                     |

---

|               |                      |
|---------------|----------------------|
| bind_remstats | <i>bind_remstats</i> |
|---------------|----------------------|

---

**Description**

See [bind\\_remstats](#).

**Usage**

bind\_remstats(...)

**Arguments**

- |     |                                                                                                                                                                                                                                 |
|-----|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| ... | Any number of <a href="#">remstats</a> objects. All the remstats objects must have matching dimensions, except for the third dimension. Note that duplicated statistics in the combined object are removed based on their name. |
|-----|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|

---

degreeDiff

*degreeDiff*


---

### Description

See [degreeDiff](#).

### Usage

```
degreeDiff(scaling = c("none", "std"), consider_type = "ignore")
```

### Arguments

|               |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              |
|---------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| scaling       | the method for scaling the degree statistic. Default is to not scale the statistic (scaling = "none"). Alternatively, standardization of the degree difference per time point can be requested with 'std'.                                                                                                                                                                                                                                                                                                   |
| consider_type | character. Controls how event types are handled: "ignore" (default): aggregate over all event types (one statistic); "separate": compute C type-specific statistics, where the type-c statistic for a dyad reflects past type-c events on that actor pair ; "interact": compute C^2 statistics capturing past-event-type x dyad-type interactions (only meaningful with extend_riskset_by_type=TRUE in remify object). Also accepts FALSE (-> "ignore") and TRUE (-> "separate") for backward compatibility. |

---

degreeMax

*degreeMax*


---

### Description

See [degreeMax](#).

### Usage

```
degreeMax(scaling = c("none", "prop", "std"), consider_type = "ignore")
```

### Arguments

|         |                                                                                                                                                                                                                                                                                                                              |
|---------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| scaling | the method for scaling the degree statistic. Default is to not scale the statistic (scaling = "none"). Alternatively, scaling of the raw degree counts by two times the number of past events at time t can be requested with 'prop' or standardization of the raw degree counts per time point can be requested with 'std'. |
|---------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|

consider\_type character. Controls how event types are handled: "ignore" (default): aggregate over all event types (one statistic); "separate": compute C type-specific statistics, where the type-c statistic for a dyad reflects past type-c events on that actor pair ; "interact": compute C^2 statistics capturing past-event-type x dyad-type interactions (only meaningful with extend\_riskset\_by\_type=TRUE in remify object). Also accepts FALSE (-> "ignore") and TRUE (-> "separate") for backward compatibility.

---

|           |                  |
|-----------|------------------|
| degreeMin | <i>degreeMin</i> |
|-----------|------------------|

---

**Description**

See [degreeMin](#).

**Usage**

degreeMin(scaling = c("none", "prop", "std"), consider\_type = "ignore")

**Arguments**

scaling the method for scaling the degree statistic. Default is to not scale the statistic (scaling = "none"). Alternatively, scaling of the raw degree counts by two times the number of past events at time t can be requested with 'prop' or standardization of the raw degree counts per time point can be requested with 'std'.

consider\_type character. Controls how event types are handled: "ignore" (default): aggregate over all event types (one statistic); "separate": compute C type-specific statistics, where the type-c statistic for a dyad reflects past type-c events on that actor pair ; "interact": compute C^2 statistics capturing past-event-type x dyad-type interactions (only meaningful with extend\_riskset\_by\_type=TRUE in remify object). Also accepts FALSE (-> "ignore") and TRUE (-> "separate") for backward compatibility.

---

|             |                    |
|-------------|--------------------|
| diagnostics | <i>diagnostics</i> |
|-------------|--------------------|

---

**Description**

See [diagnostics](#).

**Usage**

diagnostics(object, reh, stats, ...)

**Arguments**

|        |                                                                  |
|--------|------------------------------------------------------------------|
| object | is a remstimate object.                                          |
| reh    | is a remify object, the same used for the 'remstimate' object.   |
| stats  | is a remstats object, the same used for the 'remstimate' object. |
| ...    | further arguments to be passed to the 'diagnostics' method.      |

---

|            |                   |
|------------|-------------------|
| difference | <i>difference</i> |
|------------|-------------------|

---

**Description**

See [difference](#).

**Usage**

```
difference(
  variable,
  attr_actors = NULL,
  scaling = c("none", "std"),
  absolute = TRUE,
  attr_data
)
```

**Arguments**

|             |                                                                                                                                                                          |
|-------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| variable    | string with the name of the column in the attr_actors object for which the statistic has to be computed.                                                                 |
| attr_actors | optionally, an object of class <a href="#">data.frame</a> that contains the attribute, see 'Details.'                                                                    |
| scaling     | the method for scaling the statistic. Default is to not scale the statistic. Alternatively, standardization of the statistic per time point can be requested with "std". |
| absolute    | Logical value indicating whether the difference values should be converted to the absolute difference (default is TRUE).                                                 |
| attr_data   | Deprecated argument. Please use 'attr_actors' instead.                                                                                                                   |

dlcrem

*dlcrem***Description**

See [dlcrem](#).

**Usage**

```
dlcrem(reh, stats, k = 2L, nrep = 3L, ...)
```

**Arguments**

|       |                                                              |
|-------|--------------------------------------------------------------|
| reh   | A remify object.                                             |
| stats | A remstats object.                                           |
| k     | Number of latent classes (default 2).                        |
| nrep  | Number of random restarts (default 3).                       |
| ...   | Additional arguments passed to <a href="#">remestimate</a> . |

dyad

*dyad***Description**

See [dyad](#).

**Usage**

```
dyad(variable, attr_dyads = NULL, scaling = c("none", "std"), x, variableName)
```

**Arguments**

|            |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |
|------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| variable   | A string specifying the attribute to compute the statistic. If <code>attr_dyads</code> is a <code>data.frame</code> , this refers to the column name in <code>attr_actors</code> . If <code>attr_dyads</code> is a <code>matrix</code> , this corresponds to the name of the exogenous attribute, used to label the statistic in the resulting <code>remstats</code> object.                                                                                                                                                                                                                                   |
| attr_dyads | A <code>data.frame</code> or <code>matrix</code> containing attribute information for dyads. If <code>attr_dyads</code> is a <code>data.frame</code> , the first two columns should represent "actor1" and "actor2" (for directed events, "actor1" corresponds to the sender, and "actor2" corresponds to the receiver). Additional columns can represent dyads' exogenous attributes. If attributes vary over time, include a column named "time". If <code>attr_dyads</code> is a <code>matrix</code> , the rows correspond to "actor1", columns to "actor2", and cells contain dyads' exogenous attributes. |

|              |                                                                                                                                                              |
|--------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------|
| scaling      | The method for scaling the statistic. The default is no scaling. Alternatively, standardization of the statistic per time point can be requested with "std". |
| x            | Deprecated argument. Please use 'attr_dyads' instead.                                                                                                        |
| variableName | Deprecated argument. Please use 'variable' instead.                                                                                                          |

---

|                   |                                                       |
|-------------------|-------------------------------------------------------|
| edgelist_duration | <i>Example relational event history with duration</i> |
|-------------------|-------------------------------------------------------|

---

**Description**

A small directed relational event history where each event has a known start and end time, suitable for duration modeling.

**Usage**

```
data(edgelist_duration)
```

**Format**

A data frame with columns:

- time** Event start time.
- actor1** Sender.
- actor2** Receiver.
- end\_time** Event end time.

**Examples**

```
data(edgelist_duration)
reh <- remify(edgelist_duration, duration = TRUE)
```

---

|       |              |
|-------|--------------|
| event | <i>event</i> |
|-------|--------------|

---

**Description**

See [event](#).

**Usage**

```
event(variable, event_attr)
```

**Arguments**

- variable string with the name of the column in the event\_attr object for which the statistic has to be computed.
- event\_attr an object of class data.frame that contains the attribute

---

|       |              |
|-------|--------------|
| FType | <i>FType</i> |
|-------|--------------|

---

**Description**

See [FType](#).

**Usage**

FType()

---

|             |                    |
|-------------|--------------------|
| frailty_rem | <i>frailty_rem</i> |
|-------------|--------------------|

---

**Description**

See [frailty\\_rem](#).

**Usage**

frailty\_rem(reh, stats, ...)

**Arguments**

- |       |                                                             |
|-------|-------------------------------------------------------------|
| reh   | A remify or remify_durem object.                            |
| stats | A remstats object (tomstats, aomstats, or remstats_durem).  |
| ...   | Additional arguments passed to <a href="#">remstimate</a> . |

---

|               |                      |
|---------------|----------------------|
| history_durem | <i>history_durem</i> |
|---------------|----------------------|

---

**Description**

A duration-extended version of the history dataset. Each row is one directed interaction among 10 actors with a start time (time) and end time (end). Right-censored events have end = NA.

**Usage**

history\_durem

**Format**

A data frame with 115 rows and 6 columns:

**time** Start time of the event (seconds)

**actor1** Initiating actor ID

**actor2** Receiving actor ID

**setting** Context: "work" or "social"

**weight** Intrinsic event weight

**end** End time of the event (NA = right-censored)

---

|                  |                         |
|------------------|-------------------------|
| indegreeReceiver | <i>indegreeReceiver</i> |
|------------------|-------------------------|

---

**Description**

See [indegreeReceiver](#).

**Usage**

```
indegreeReceiver(scaling = c("none", "prop", "std"), consider_type = "ignore")
```

**Arguments**

**scaling** the method for scaling the degree statistic. Default is to not scale the statistic (scaling = "none"). Alternatively, scaling of the raw degree counts by the number of past events at time t can be requested with 'prop' or standardization of the raw degree counts per time point can be requested with 'std'.

**consider\_type** character. Controls how event types are handled: "ignore" (default): aggregate over all event types (one statistic); "separate": compute C type-specific statistics, where the type-c statistic for a dyad reflects past type-c events on that actor pair ; "interact": compute C^2 statistics capturing past-event-type x dyad-type interactions (only meaningful with extend\_riskset\_by\_type=TRUE in remify object). Also accepts FALSE (-> "ignore") and TRUE (-> "separate") for backward compatibility.

---

|                |                       |
|----------------|-----------------------|
| indegreeSender | <i>indegreeSender</i> |
|----------------|-----------------------|

---

**Description**

See [indegreeSender](#).

**Usage**

```
indegreeSender(scaling = c("none", "prop", "std"), consider_type = "ignore")
```

**Arguments**

|               |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              |
|---------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| scaling       | the method for scaling the degree statistic. Default is to not scale the statistic (scaling = "none"). Alternatively, scaling of the raw degree counts by the number of past events at time t can be requested with 'prop' or standardization of the raw degree counts per time point can be requested with 'std'.                                                                                                                                                                                           |
| consider_type | character. Controls how event types are handled: "ignore" (default): aggregate over all event types (one statistic); "separate": compute C type-specific statistics, where the type-c statistic for a dyad reflects past type-c events on that actor pair ; "interact": compute C^2 statistics capturing past-event-type x dyad-type interactions (only meaningful with extend_riskset_by_type=TRUE in remify object). Also accepts FALSE (-> "ignore") and TRUE (-> "separate") for backward compatibility. |

---

|         |                |
|---------|----------------|
| inertia | <i>inertia</i> |
|---------|----------------|

---

**Description**

See [inertia](#).

**Usage**

```
inertia(scaling = c("none", "prop", "std"), consider_type = "ignore")
```

**Arguments**

|         |                                                                                                                                                                                                                                                                                                                                                     |
|---------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| scaling | the method for scaling the inertia statistic. Default is to not scale the statistic (scaling = "none"). Alternatively, the statistics can be scaled by specifying 'prop', in which raw counts are divided by the outdegree of the sender at time t (see 'details') or standardization of the raw counts per time point can be requested with 'std'. |
|---------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|

**consider\_type** character. Controls how event types are handled: "ignore" (default): aggregate over all event types (one statistic); "separate": compute C type-specific statistics, where the type-c statistic for a dyad reflects past type-c events on that actor pair ; "interact": compute C^2 statistics capturing past-event-type x dyad-type interactions (only meaningful with extend\_riskset\_by\_type=TRUE in remify object). Also accepts FALSE (-> "ignore") and TRUE (-> "separate") for backward compatibility.

---

|       |                                          |
|-------|------------------------------------------|
| info3 | <i>Exogenous information of 5 actors</i> |
|-------|------------------------------------------|

---

**Description**

A dataset containing the (hypothetical) age of the actors in a social network of a relational event history [randomREH3](#).

**Usage**

data(info3)

**Format**

- A dataframe with 7 rows and 4 variables:
- name** numeric id of the actor
  - time** numeric value, describes when the value of the covariate changes, if it changes
  - age** age of the actor
  - sex** dichotomized sex of the actor (e.g., 0 = male, 1 = female)

**See Also**

[randomREH3](#) for the relational event history.

---

|                 |                        |
|-----------------|------------------------|
| is.remify_durem | <i>is.remify_durem</i> |
|-----------------|------------------------|

---

**Description**

See [is.remify\\_durem](#).

**Usage**

is.remify\_durem(x)

**Arguments**

x                      Any R object.

---

|                   |                          |
|-------------------|--------------------------|
| is.remstats_durem | <i>is.remstats_durem</i> |
|-------------------|--------------------------|

---

**Description**

See [is.remstats\\_durem](#).

**Usage**

```
is.remstats_durem(x)
```

**Arguments**

|   |               |
|---|---------------|
| x | Any R object. |
|---|---------------|

---

|     |            |
|-----|------------|
| isp | <i>isp</i> |
|-----|------------|

---

**Description**

See [isp](#).

**Usage**

```
isp(unique = FALSE, scaling = c("none", "std"), consider_type = "ignore")
```

**Arguments**

|               |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              |
|---------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| unique        | A logical value indicating whether to sum the minimum of events with third actors (FALSE, default) or the number of third actors that create a new, unique shared partner (TRUE). See details for more information.                                                                                                                                                                                                                                                                                          |
| scaling       | the method for scaling the triad statistic. Default is to not scale the statistic but keep the raw 'counts'. Alternatively, standardization of the raw counts per time point can be requested with 'std'.                                                                                                                                                                                                                                                                                                    |
| consider_type | character. Controls how event types are handled: "ignore" (default): aggregate over all event types (one statistic); "separate": compute C type-specific statistics, where the type-c statistic for a dyad reflects past type-c events on that actor pair ; "interact": compute C^2 statistics capturing past-event-type x dyad-type interactions (only meaningful with extend_riskset_by_type=TRUE in remify object). Also accepts FALSE (-> "ignore") and TRUE (-> "separate") for backward compatibility. |

---

|     |            |
|-----|------------|
| itp | <i>itp</i> |
|-----|------------|

---

**Description**

See [itp](#).

**Usage**

```
itp(unique = FALSE, scaling = c("none", "std"), consider_type = "ignore")
```

**Arguments**

- unique            A logical value indicating whether to sum the minimum of events with third actors (FALSE, default) or the number of third actors that create a new, unique two-path (TRUE). See details for more information.
- scaling           The method for scaling the triad statistic. The default value is "none", which means the statistic is not scaled. Alternatively, you can set it to "std" to request standardization of the raw counts per time point.
- consider\_type    character. Controls how event types are handled: "ignore" (default): aggregate over all event types (one statistic); "separate": compute C type-specific statistics, where the type-c statistic for a dyad reflects past type-c events on that actor pair ; "interact": compute C^2 statistics capturing past-event-type x dyad-type interactions (only meaningful with extend\_riskset\_by\_type=TRUE in remify object). Also accepts FALSE (-> "ignore") and TRUE (-> "separate") for backward compatibility.

---

|         |                |
|---------|----------------|
| maximum | <i>maximum</i> |
|---------|----------------|

---

**Description**

See [maximum](#).

**Usage**

```
maximum(variable, attr_actors = NULL, scaling = c("none", "std"), attr_data)
```

**Arguments**

- variable           string with the name of the column in the attr\_actors object for which the statistic has to be computed.
- attr\_actors        optionally, an object of class [data.frame](#) that contains the attribute, see 'Details.'

|           |                                                                                                                                                                          |
|-----------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| scaling   | the method for scaling the statistic. Default is to not scale the statistic. Alternatively, standardization of the statistic per time point can be requested with "std". |
| attr_data | Deprecated argument. Please use 'attr_actors' instead.                                                                                                                   |

---

|         |                |
|---------|----------------|
| minimum | <i>minimum</i> |
|---------|----------------|

---

### Description

See [minimum](#).

### Usage

```
minimum(variable, attr_actors = NULL, scaling = c("none", "std"), attr_data)
```

### Arguments

|             |                                                                                                                                                                          |
|-------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| variable    | string with the name of the column in the attr_actors object for which the statistic has to be computed.                                                                 |
| attr_actors | optionally, an object of class <a href="#">data.frame</a> that contains the attribute, see 'Details.'                                                                    |
| scaling     | the method for scaling the statistic. Default is to not scale the statistic. Alternatively, standardization of the statistic per time point can be requested with "std". |
| attr_data   | Deprecated argument. Please use 'attr_actors' instead.                                                                                                                   |

---

|     |            |
|-----|------------|
| osp | <i>osp</i> |
|-----|------------|

---

### Description

See [osp](#).

### Usage

```
osp(unique = FALSE, scaling = c("none", "std"), consider_type = "ignore")
```

Arguments

|               |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              |
|---------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| unique        | A logical value indicating whether to sum the minimum of events with third actors (FALSE, default) or the number of third actors that create a new, unique shared partner (TRUE). See details for more information.                                                                                                                                                                                                                                                                                          |
| scaling       | the method for scaling the triad statistic. Default is to not scale the statistic but keep the raw 'counts'. Alternatively, standardization of the raw counts per time point can be requested with 'std'.                                                                                                                                                                                                                                                                                                    |
| consider_type | character. Controls how event types are handled: "ignore" (default): aggregate over all event types (one statistic); "separate": compute C type-specific statistics, where the type-c statistic for a dyad reflects past type-c events on that actor pair ; "interact": compute C^2 statistics capturing past-event-type x dyad-type interactions (only meaningful with extend_riskset_by_type=TRUE in remify object). Also accepts FALSE (-> "ignore") and TRUE (-> "separate") for backward compatibility. |

---

|     |            |
|-----|------------|
| otp | <i>otp</i> |
|-----|------------|

---

Description

See [otp](#).

Usage

```
otp(unique = FALSE, scaling = c("none", "std"), consider_type = "ignore")
```

Arguments

|               |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              |
|---------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| unique        | A logical value indicating whether to sum the minimum of events with third actors (FALSE, default) or the number of third actors that create a new, unique two-path (TRUE). See details for more information.                                                                                                                                                                                                                                                                                                |
| scaling       | The method for scaling the triad statistic. The default value is "none", which means the statistic is not scaled. Alternatively, you can set it to "std" to request standardization of the raw counts per time point.                                                                                                                                                                                                                                                                                        |
| consider_type | character. Controls how event types are handled: "ignore" (default): aggregate over all event types (one statistic); "separate": compute C type-specific statistics, where the type-c statistic for a dyad reflects past type-c events on that actor pair ; "interact": compute C^2 statistics capturing past-event-type x dyad-type interactions (only meaningful with extend_riskset_by_type=TRUE in remify object). Also accepts FALSE (-> "ignore") and TRUE (-> "separate") for backward compatibility. |

---

|                   |                          |
|-------------------|--------------------------|
| outdegreeReceiver | <i>outdegreeReceiver</i> |
|-------------------|--------------------------|

---

**Description**

See [outdegreeReceiver](#).

**Usage**

```
outdegreeReceiver(scaling = c("none", "prop", "std"), consider_type = "ignore")
```

**Arguments**

|               |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              |
|---------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| scaling       | the method for scaling the degree statistic. Default is to not scale the statistic (scaling = "none"). Alternatively, scaling of the raw degree counts by the number of past events at time t can be requested with 'prop' or standardization of the raw degree counts per time point can be requested with 'std'.                                                                                                                                                                                           |
| consider_type | character. Controls how event types are handled: "ignore" (default): aggregate over all event types (one statistic); "separate": compute C type-specific statistics, where the type-c statistic for a dyad reflects past type-c events on that actor pair ; "interact": compute C^2 statistics capturing past-event-type x dyad-type interactions (only meaningful with extend_riskset_by_type=TRUE in remify object). Also accepts FALSE (-> "ignore") and TRUE (-> "separate") for backward compatibility. |

---

|                 |                        |
|-----------------|------------------------|
| outdegreeSender | <i>outdegreeSender</i> |
|-----------------|------------------------|

---

**Description**

See [outdegreeSender](#).

**Usage**

```
outdegreeSender(scaling = c("none", "prop", "std"), consider_type = "ignore")
```

**Arguments**

|         |                                                                                                                                                                                                                                                                                                                    |
|---------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| scaling | the method for scaling the degree statistic. Default is to not scale the statistic (scaling = "none"). Alternatively, scaling of the raw degree counts by the number of past events at time t can be requested with 'prop' or standardization of the raw degree counts per time point can be requested with 'std'. |
|---------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|

`consider_type` character. Controls how event types are handled: "ignore" (default): aggregate over all event types (one statistic); "separate": compute C type-specific statistics, where the type-c statistic for a dyad reflects past type-c events on that actor pair ; "interact": compute C^2 statistics capturing past-event-type x dyad-type interactions (only meaningful with `extend_riskset_by_type=TRUE` in `remify` object). Also accepts FALSE (-> "ignore") and TRUE (-> "separate") for backward compatibility.

---

|                    |              |
|--------------------|--------------|
| <code>psABA</code> | <i>psABA</i> |
|--------------------|--------------|

---

**Description**

See [psABA](#).

**Usage**

`psABA()`

---

|                     |               |
|---------------------|---------------|
| <code>psABAB</code> | <i>psABAB</i> |
|---------------------|---------------|

---

**Description**

See [psABAB](#).

**Usage**

`psABAB(consider_type = "ignore")`

**Arguments**

`consider_type` character. Controls how event types are handled: "ignore" (default): aggregate over all event types (one statistic); "separate": compute C type-specific statistics, where the type-c statistic for a dyad reflects past type-c events on that actor pair ; "interact": compute C^2 statistics capturing past-event-type x dyad-type interactions (only meaningful with `extend_riskset_by_type=TRUE` in `remify` object). Also accepts FALSE (-> "ignore") and TRUE (-> "separate") for backward compatibility.

---

psABAY

*psABAY*


---

**Description**

See [psABAY](#).

**Usage**

```
psABAY(consider_type = "ignore")
```

**Arguments**

**consider\_type** character. Controls how event types are handled: "ignore" (default): aggregate over all event types (one statistic); "separate": compute C type-specific statistics, where the type-c statistic for a dyad reflects past type-c events on that actor pair ; "interact": compute C^2 statistics capturing past-event-type x dyad-type interactions (only meaningful with extend\_riskset\_by\_type=TRUE in remify object). Also accepts FALSE (-> "ignore") and TRUE (-> "separate") for backward compatibility.

---

psABB

*psABB*


---

**Description**

See [psABB](#).

**Usage**

```
psABB()
```

---

psABBA

*psABBA*


---

**Description**

See [psABBA](#).

**Usage**

```
psABBA(consider_type = "ignore")
```

Arguments

consider\_type character. Controls how event types are handled: "ignore" (default): aggregate over all event types (one statistic); "separate": compute C type-specific statistics, where the type-c statistic for a dyad reflects past type-c events on that actor pair ; "interact": compute C^2 statistics capturing past-event-type x dyad-type interactions (only meaningful with extend\_riskset\_by\_type=TRUE in remify object). Also accepts FALSE (-> "ignore") and TRUE (-> "separate") for backward compatibility.

---

|        |        |
|--------|--------|
| psABBY | psABBY |
|--------|--------|

---

Description

See [psABBY](#).

Usage

```
psABBY(consider_type = "ignore")
```

Arguments

consider\_type character. Controls how event types are handled: "ignore" (default): aggregate over all event types (one statistic); "separate": compute C type-specific statistics, where the type-c statistic for a dyad reflects past type-c events on that actor pair ; "interact": compute C^2 statistics capturing past-event-type x dyad-type interactions (only meaningful with extend\_riskset\_by\_type=TRUE in remify object). Also accepts FALSE (-> "ignore") and TRUE (-> "separate") for backward compatibility.

---

|       |       |
|-------|-------|
| psABX | psABX |
|-------|-------|

---

Description

See [psABX](#).

Usage

```
psABX()
```

---

psABXA

---

*psABXA*


---

**Description**

See [psABXA](#).

**Usage**

```
psABXA(consider_type = "ignore")
```

**Arguments**

**consider\_type** character. Controls how event types are handled: "ignore" (default): aggregate over all event types (one statistic); "separate": compute C type-specific statistics, where the type-c statistic for a dyad reflects past type-c events on that actor pair ; "interact": compute  $C^2$  statistics capturing past-event-type x dyad-type interactions (only meaningful with `extend_riskset_by_type=TRUE` in `remify` object). Also accepts FALSE (-> "ignore") and TRUE (-> "separate") for backward compatibility.

---

psABXB

---

*psABXB*


---

**Description**

See [psABXB](#).

**Usage**

```
psABXB(consider_type = "ignore")
```

**Arguments**

**consider\_type** character. Controls how event types are handled: "ignore" (default): aggregate over all event types (one statistic); "separate": compute C type-specific statistics, where the type-c statistic for a dyad reflects past type-c events on that actor pair ; "interact": compute  $C^2$  statistics capturing past-event-type x dyad-type interactions (only meaningful with `extend_riskset_by_type=TRUE` in `remify` object). Also accepts FALSE (-> "ignore") and TRUE (-> "separate") for backward compatibility.

---

|        |               |
|--------|---------------|
| psABXY | <i>psABXY</i> |
|--------|---------------|

---

**Description**

See [psABXY](#).

**Usage**

```
psABXY(consider_type = "ignore")
```

**Arguments**

consider\_type    character. Controls how event types are handled: "ignore" (default): aggregate over all event types (one statistic); "separate": compute C type-specific statistics, where the type-c statistic for a dyad reflects past type-c events on that actor pair ; "interact": compute C^2 statistics capturing past-event-type x dyad-type interactions (only meaningful with extend\_riskset\_by\_type=TRUE in remify object). Also accepts FALSE (-> "ignore") and TRUE (-> "separate") for backward compatibility.

---

|           |                  |
|-----------|------------------|
| randomREH | <i>randomREH</i> |
|-----------|------------------|

---

**Description**

See [randomREH](#).

---

|            |                                                                           |
|------------|---------------------------------------------------------------------------|
| randomREH3 | <i>Generated Relational Event History with Duration, Type, and Weight</i> |
|------------|---------------------------------------------------------------------------|

---

**Description**

A randomly generated sequence of 999 duration relational events among 5 actors. Starting an event depended on inertia, reciprocity, sender out-degree, incoming shared partners, and sender age; (undirected) ending an event depended on the dyad's total active degree and the age difference of the two actors. Exponential memory decay was used with a half-life of 2000. Actor covariates are in [info3](#).

**Usage**

```
data(randomREH3)
```

**Format**

`data(randomREH3)` loads a `data.frame` with 999 rows and 6 columns:

`time` the timestamp indicating when each event started;

`actor1` the actor that generated (initiated) the event;

`actor2` the actor that received the event;

`end` the time at which the event ended (NA = right-censored);

`setting` the setting (type) of the event, either `social` or `work`;

`duration` the duration of each event (`end - time`), which can also serve as an event weight.

**See Also**

[info3](#) for actor covariates.

---

|                      |                |
|----------------------|----------------|
| <code>receive</code> | <i>receive</i> |
|----------------------|----------------|

---

**Description**

See [receive](#).

**Usage**

```
receive(variable, attr_actors = NULL, scaling = c("none", "std"), attr_data)
```

**Arguments**

- |                          |                                                                                                                                                                          |
|--------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <code>variable</code>    | string with the name of the column in the <code>attr_actors</code> object for which the statistic has to be computed.                                                    |
| <code>attr_actors</code> | optionally, an object of class <a href="#">data.frame</a> that contains the attribute, see 'Details.'                                                                    |
| <code>scaling</code>     | the method for scaling the statistic. Default is to not scale the statistic. Alternatively, standardization of the statistic per time point can be requested with "std". |
| <code>attr_data</code>   | Deprecated argument. Please use 'attr_actors' instead.                                                                                                                   |

---

|                 |                        |
|-----------------|------------------------|
| recencyContinue | <i>recencyContinue</i> |
|-----------------|------------------------|

---

**Description**

See [recencyContinue](#).

**Usage**

```
recencyContinue(consider_type = "ignore")
```

**Arguments**

**consider\_type** character. Controls how event types are handled: "ignore" (default): aggregate over all event types (one statistic); "separate": compute C type-specific statistics, where the type-c statistic for a dyad reflects past type-c events on that actor pair ; "interact": compute  $C^2$  statistics capturing past-event-type x dyad-type interactions (only meaningful with `extend_riskset_by_type=TRUE` in `remify` object). Also accepts FALSE (-> "ignore") and TRUE (-> "separate") for backward compatibility.

---

|                        |                               |
|------------------------|-------------------------------|
| recencyReceiveReceiver | <i>recencyReceiveReceiver</i> |
|------------------------|-------------------------------|

---

**Description**

See [recencyReceiveReceiver](#).

**Usage**

```
recencyReceiveReceiver(consider_type = "ignore")
```

**Arguments**

**consider\_type** character. Controls how event types are handled: "ignore" (default): aggregate over all event types (one statistic); "separate": compute C type-specific statistics, where the type-c statistic for a dyad reflects past type-c events on that actor pair ; "interact": compute  $C^2$  statistics capturing past-event-type x dyad-type interactions (only meaningful with `extend_riskset_by_type=TRUE` in `remify` object). Also accepts FALSE (-> "ignore") and TRUE (-> "separate") for backward compatibility.

---

recencyReceiveSender    *recencyReceiveSender*


---

**Description**

See [recencyReceiveSender](#).

**Usage**

```
recencyReceiveSender(consider_type = "ignore")
```

**Arguments**

**consider\_type**    character. Controls how event types are handled: "ignore" (default): aggregate over all event types (one statistic); "separate": compute C type-specific statistics, where the type-c statistic for a dyad reflects past type-c events on that actor pair ; "interact": compute  $C^2$  statistics capturing past-event-type x dyad-type interactions (only meaningful with `extend_riskset_by_type=TRUE` in `remify` object). Also accepts FALSE (-> "ignore") and TRUE (-> "separate") for backward compatibility.

---

recencySendReceiver    *recencySendReceiver*


---

**Description**

See [recencySendReceiver](#).

**Usage**

```
recencySendReceiver(consider_type = "ignore")
```

**Arguments**

**consider\_type**    character. Controls how event types are handled: "ignore" (default): aggregate over all event types (one statistic); "separate": compute C type-specific statistics, where the type-c statistic for a dyad reflects past type-c events on that actor pair ; "interact": compute  $C^2$  statistics capturing past-event-type x dyad-type interactions (only meaningful with `extend_riskset_by_type=TRUE` in `remify` object). Also accepts FALSE (-> "ignore") and TRUE (-> "separate") for backward compatibility.

---

|                   |                          |
|-------------------|--------------------------|
| recencySendSender | <i>recencySendSender</i> |
|-------------------|--------------------------|

---

### Description

See [recencySendSender](#).

### Usage

```
recencySendSender(consider_type = "ignore")
```

### Arguments

**consider\_type** character. Controls how event types are handled: "ignore" (default): aggregate over all event types (one statistic); "separate": compute C type-specific statistics, where the type-c statistic for a dyad reflects past type-c events on that actor pair ; "interact": compute C^2 statistics capturing past-event-type x dyad-type interactions (only meaningful with `extend_riskset_by_type=TRUE` in `remify` object). Also accepts FALSE (-> "ignore") and TRUE (-> "separate") for backward compatibility.

---

|             |                    |
|-------------|--------------------|
| reciprocity | <i>reciprocity</i> |
|-------------|--------------------|

---

### Description

See [reciprocity](#).

### Usage

```
reciprocity(scaling = c("none", "prop", "std"), consider_type = "ignore")
```

### Arguments

**scaling** the method for scaling the reciprocity statistic. Default is to not scale the statistic but keep the raw 'counts'. Alternatively, the statistics can be scaled by 'prop', in which raw counts are divided by the indegree of the sender at time t (see 'details') or standardization of the raw counts per time point can be requested with 'std'.

**consider\_type** character. Controls how event types are handled: "ignore" (default): aggregate over all event types (one statistic); "separate": compute C type-specific statistics, where the type-c statistic for a dyad reflects past type-c events on that actor pair ; "interact": compute C^2 statistics capturing past-event-type x dyad-type interactions (only meaningful with `extend_riskset_by_type=TRUE` in `remify` object). Also accepts FALSE (-> "ignore") and TRUE (-> "separate") for backward compatibility.

---

|            |                   |
|------------|-------------------|
| remfrailty | <i>remfrailty</i> |
|------------|-------------------|

---

**Description**

See [remfrailty](#).

**Usage**

```
remfrailty(
  reh,
  stats,
  approach = c("frequentist", "Bayesian"),
  engine = "auto",
  ...
)
```

**Arguments**

|          |                                                                                                                                                                                       |
|----------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| reh      | A remify or remify_durem object.                                                                                                                                                      |
| stats    | A remstats object (tomstats, aomstats, or remstats_durem).                                                                                                                            |
| approach | Either "frequentist" or "Bayesian". Bayesian frailty is not offered in this version.                                                                                                  |
| engine   | For frequentist, the interval-model backend: "glmmTMB" or "lme4". The default "auto" uses glmmTMB when installed, otherwise lme4. Ignored for ordinal models, which always use coxme. |
| ...      | Additional arguments passed to <a href="#">remstimate</a> .                                                                                                                           |

---

|        |               |
|--------|---------------|
| remify | <i>remify</i> |
|--------|---------------|

---

**Description**

See [remify](#).

**Usage**

```
remify(
  edgelist,
  directed = TRUE,
  ordinal = FALSE,
  model = c("tie", "actor"),
  actors = NULL,
  riskset = c("full", "active", "active_saturated", "manual"),
```

```

manual_riskset = NULL,
extend_riskset_by_type = FALSE,
event_type = NULL,
event_weight = NULL,
origin = NULL,
time_units = c("auto", "secs", "mins", "hours", "days", "weeks", "months", "years"),
aggregate_time = 1,
attach_riskset = TRUE,
riskset_decode = c("labels", "ids", "none"),
riskset_max_decode = 200000L,
event_attributes = NULL,
ncores = 1L,
duration = FALSE,
dur_directed_end = FALSE,
dur_type_exclusive = FALSE
)

```

## Arguments

|          |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            |
|----------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| edgelist | the relational event history. An object of class <code>data.frame</code> with first three columns corresponding to time, and actors forming the dyad. The first three columns will be re-named "time", "actor1", "actor2" (where, for directed networks, "actor1" corresponds to the sender and "actor2" to the receiver of the relational event). Optional columns that can be supplied are: 'type' and 'weight'. If one or both exist in edgelist, they have to be named accordingly.                                                                                                                                                                    |
| directed | logical value indicating whether events are directed (TRUE) or undirected (FALSE). (default value is TRUE)                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 |
| ordinal  | logical value indicating whether only the order of events matters in the model (TRUE) or also the exact timing must be considered in the model (FALSE). (default value is FALSE). If TRUE, then the column "time" of edgelist is still used to extract the order.                                                                                                                                                                                                                                                                                                                                                                                          |
| model    | either "tie" (default) or "actor" oriented modeling. For "tie", the riskset is at the dyad level. For "actor", the model has two sub-processes: a sender rate model (who sends next?) and a receiver choice model (who does the sender choose?). Actor-oriented modeling requires directed=TRUE. The returned object includes sender_riskset, receiver_riskset, and activeN (see @return).                                                                                                                                                                                                                                                                 |
| actors   | [ <i>optional</i> ] character vector of actors' names that may be observed interacting in the network. If NULL (default), actors' names will be taken from the input edgelist.                                                                                                                                                                                                                                                                                                                                                                                                                                                                             |
| riskset  | [ <i>optional</i> ] character value indicating the type of risk set to process: riskset = "full" (default) consists of all the possible dyadic events given the number of actors (and the number of event types) and it maintains the same structure over time. riskset = "active" considers at risk only the observed dyads and it maintains the same structure over time. riskset = "manual", allows the risk set to have a structure that is user-defined, and it is based on the instructions supplied via the argument manual_riskset. riskset = "active_saturated" extends the active riskset by adding the reverse direction for each observed dyad |

|                        |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              |
|------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|                        | (if A->B is observed, B->A is also at risk) and includes all event types for each observed actor pair (type column is ignored). This reflects the assumption that observing any interaction between two actors implies both directions and all types are possible.                                                                                                                                                                                                                                                                                                                                                                                           |
| manual_riskset         | [ <i>optional</i> ] When riskset = "manual", this argument of class <code>data.frame</code> specifies which dyadic riskset to consider through the entire sequence. If observed dyads from the edgelist are missing, they will be automatically be added.                                                                                                                                                                                                                                                                                                                                                                                                    |
| extend_riskset_by_type | logical. FALSE (default). When event types are present (via event_type), controls whether the risk set is expanded over types. If TRUE (default when types are present), each actor pair is duplicated for each event type, so the risk set has size $D = N(N - 1) \times C$ (directed) or $D = N(N - 1)/2 \times C$ (undirected), and the type column appears in the decoded risk set. If FALSE, event type is treated as a mark on events only and does not expand the risk set: $D = N(N - 1)$ (directed) or $D = N(N - 1)/2$ (undirected), and no type column appears in the decoded risk set. This argument is ignored when no event types are present. |
| event_type             | Optional. Either NULL (default) or a single character string giving the name of the column in edgelist that contains event types (marks).<br>If event_type is NULL, remify() uses edgelist\$type if it exists; otherwise events are treated as untyped.<br>If event_type is a column name, that column is used as the event-type mark. If a column named type already exists and event_type != "type", the existing edgelist\$type is overridden (with a warning).<br>When event types are present (via edgelist\$type or event_type), the dyadic risk set is extended over types, i.e., each dyad is duplicated for each event type (dyad $\times$ type).   |
| event_weight           | Optional. Either NULL (default) or a single character string giving the name of the column in edgelist that contains event weights.<br>If event_weight is NULL, remify() uses edgelist\$weight if it exists; otherwise events are treated as unweighted (uniform weight = 1).<br>If event_weight is a column name, that column is used as the event weight. If a column named weight already exists and event_weight != "weight", the existing edgelist\$weight is overridden (with a warning). This argument behaves analogously to event_type and applies to both standard and duration (duration = TRUE) models.                                          |
| origin                 | [ <i>optional</i> ] starting time point of the observation period (default is NULL). If it is supplied, it must have the same class of the 'time' column in the input edgelist. If unsupplied, the origin is set to the average waiting time in the sequence subtracted from the time of the first event.                                                                                                                                                                                                                                                                                                                                                    |
| time_units             | Character string specifying the time unit for converting time values when 'edgelist\$time' is of class Date or POSIXct; ignored for numeric or integer time. Default is "secs".                                                                                                                                                                                                                                                                                                                                                                                                                                                                              |
| aggregate_time         | Integer $\geq 1$ . Event-time aggregated based on unique time points. Keeps every aggregate_time-th unique event time (after time translation) and maps each event time to the next kept time point (i.e., ceiling to the kept grid). This reduces the number of unique time points (and thus memory/computation in later steps).                                                                                                                                                                                                                                                                                                                            |

- attach\_riskset** Logical. If TRUE, attaches a list `riskset_info` to the returned `remify` object. The list contains the effective risk set representation used for estimation (e.g., `riskset_idx`, `dyadIDactive`, dictionaries, and basic risk set metadata). This is intended to make the returned object self-describing and easier to inspect/debug.
- riskset\_decode** Character. Controls how (and whether) the included risk set dyads are decoded and attached in `riskset_info$included`.
- "labels" Attach a decoded dyad table including actor (and type) labels (e.g., `actor1`, `actor2`, and optional `type`).
  - "ids" Attach a decoded dyad table with integer IDs only (e.g., `actor1ID`, `actor2ID`, optional `typeID`, and `dyadID`).
  - "none" Do not attach a decoded dyad table.
- riskset\_max\_decode** Integer. Maximum number of included dyads (i.e., `length(riskset_idx) / D_active`) for which `riskset_decode="labels"` is allowed. If the included risk set exceeds this threshold, decoding to labels is skipped (typically falling back to "ids" with a warning) to avoid large memory usage.
- event\_attributes** Optional character vector of column names in `edgelist` to retain as additional event-level variables in the returned `reh` object.
- These columns are stored as `reh$event_attributes` together with the corresponding `time`, `actor1`, and `actor2` columns (and an internal `.event_id`). This is useful when downstream functions (e.g., in **remstats**) need access to event-level marks/attributes/covariates that are not part of the core `reh$edgelist` produced by `remify()`.
- Note: `event_attributes` does not affect risk set construction or type handling in `remify()`; it only preserves additional columns for later use. Currently there is no further support yet when `event_attributes` have been added.
- ncores** [*optional*] number of cores used in the parallelization of the processing functions. (default is 1).
- duration** Logical. If TRUE, the `edgelist` is treated as a duration `edgelist` (each event has both a start time and an end time) and a `remify_durem` object is returned instead of a standard `remify` object. The `edgelist` must contain an `end` column (or `end_time / duration` as alternatives; see `.durem_normalize_edgelist`). Default FALSE.
- dur\_directed\_end** Logical. Only used when `duration = TRUE`. If FALSE (default), the end process is undirected: either actor can terminate the event and only a combined dyad-level end rate is modelled. If TRUE, the end process is directed: a `who_ended` column in the `edgelist` ("`actor1`" / "`actor2`" / NA) records which actor terminated each event, enabling actor-level end-rate models. When `who_ended` is absent and `dur_directed_end = TRUE`, `actor1` is assumed to terminate all events and a message is issued.
- dur\_type\_exclusive** Logical. Only used when `duration = TRUE`, when typed events are present and when `extend_riskset_by_type = TRUE`. If TRUE, an active event of any type is a hard block on starting events of all other types for the same dyad (e.g. two

actors cannot start a text conversation while already in a face-to-face conversation). If FALSE (default), types are treated as independent processes and a dyad can be simultaneously active in events of different types. Has no effect when `extend_riskset_by_type = FALSE`.

---

remixture

*remixture*


---

## Description

See [remixture](#).

## Usage

```
remixture(reh, stats, random, k = 2L, concomitant = NULL, nrep = 3L, ...)
```

## Arguments

|                          |                                                                                                        |
|--------------------------|--------------------------------------------------------------------------------------------------------|
| <code>reh</code>         | A remify object.                                                                                       |
| <code>stats</code>       | A remstats object.                                                                                     |
| <code>random</code>      | Clustering formula, e.g. $\sim (1 \mid \text{dyad})$ or $\sim (1 + \text{inertia} \mid \text{dyad})$ . |
| <code>k</code>           | Number of latent classes (default 2).                                                                  |
| <code>concomitant</code> | Optional concomitant formula for class probabilities.                                                  |
| <code>nrep</code>        | Number of random restarts (default 3).                                                                 |
| <code>...</code>         | Additional arguments passed to <a href="#">remstimate</a> .                                            |

---

rempenalty

*rempenalty*


---

## Description

See [rempenalty](#).

## Usage

```
rempenalty(
  reh,
  stats,
  approach = c("frequentist", "Bayesian"),
  alpha = 1,
  prior = "horseshoe",
  nfolds = 10L,
  lambda_select = c("1se", "min"),
  ...
)
```

**Arguments**

|               |                                                                             |
|---------------|-----------------------------------------------------------------------------|
| reh           | A remify or remify_durem object.                                            |
| stats         | A remstats object.                                                          |
| approach      | "frequentist" (glmnet) or "Bayesian" (shrinkem).                            |
| alpha         | Elastic-net mixing for the frequentist fit: 1 = lasso (default), 0 = ridge. |
| prior         | Shrinkage prior for the Bayesian fit (default "horseshoe").                 |
| nfolds        | Cross-validation folds (frequentist). Default 10.                           |
| lambda_select | Which lambda (frequentist): "1se" (default) or "min".                       |
| ...           | Additional arguments passed to <a href="#">remestimate</a> .                |

---

remstats

---

*remstats*


---

**Description**

See [remstats](#).

**Usage**

```
remstats(
  reh,
  tie_effects = NULL,
  sender_effects = NULL,
  receiver_effects = NULL,
  start_effects = NULL,
  end_effects = NULL,
  memory = c("full", "window", "decay", "interval"),
  memory_value = NA,
  psi_start = 1,
  psi_end = 1,
  first = 2,
  last = Inf,
  display_progress = FALSE,
  sampling = FALSE,
  samp_num = 10L,
  seed = NULL,
  attr_actors = NULL,
  attr_dyads = NULL
)
```

**Arguments**

|                  |                                                                                                                                                                                                                                                                                                         |
|------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| reh              | an object of class " <b>remify</b> " characterizing the relational event history. May also be a <code>remify_durem</code> object for duration relational event models.                                                                                                                                  |
| tie_effects      | an object of class " <b>formula</b> " (or one that can be coerced to that class): a symbolic description of the effects in the tie-oriented model for which statistics are computed, see 'Details' for the available effects and their corresponding statistics                                         |
| sender_effects   | an object of class " <b>formula</b> " (or one that can be coerced to that class): a symbolic description of the effects in the sender activity rate step of the actor-oriented model for which statistics are computed, see 'Details'                                                                   |
| receiver_effects | an object of class " <b>formula</b> " (or one that can be coerced to that class): a symbolic description of the effects in the receiver choice step of model for which statistics are computed, see 'Details'                                                                                           |
| start_effects    | Formula for the start sub-model statistics. Only used when <code>reh</code> is a <code>remify_durem</code> object (i.e. when <code>remify(..., duration = TRUE)</code> was called). Equivalent to <code>tie_effects</code> but applied to the start process. Only supported for the tie-oriented model. |
| end_effects      | Formula for the end sub-model statistics. Only used when <code>reh</code> is a <code>remify_durem</code> object. Only supported for the tie-oriented model.                                                                                                                                             |
| memory           | The memory to be used. See 'Details'.                                                                                                                                                                                                                                                                   |
| memory_value     | Numeric value indicating the memory parameter. Default is NA, which is only valid for <code>memory = "full"</code> (no memory parameter required). See 'Details'.                                                                                                                                       |
| psi_start        | Numeric. Duration exponent for start-model history weighting. The weight of each past event in the start statistics is $\text{event\_weight} * (\text{end} - \text{time} + 1)^{\text{psi\_start}}$ . Default 1. Only used when <code>reh</code> is a <code>remify_durem</code> object.                  |
| psi_end          | Numeric. Duration exponent for end-model history weighting. The weight of each past event in the end statistics is $\text{event\_weight} * (\text{end} - \text{time} + 1)^{\text{psi\_end}}$ . Default 1. Only used when <code>reh</code> is a <code>remify_durem</code> object.                        |
| first            | an optional integer value, specifying the index of the first unique time point event in the relational event history for which statistics must be computed (see 'Details'). Default is 2: the first event has no history and is used only to initialize statistics, not to fit the model.               |
| last             | an optional integer value, specifying the index of the last unique time point in the relational event history for which statistics must be computed (see 'Details')                                                                                                                                     |
| display_progress | should a progress bar for the computation of the endogenous statistics be shown (TRUE) or not (FALSE)?                                                                                                                                                                                                  |
| sampling         | Logical. If TRUE, statistics are computed using case-control (dyad) sampling rather than the full risk set. Default FALSE. Only supported for a tie model.                                                                                                                                              |
| samp_num         | Integer. Number of dyads to include per event when <code>sampling = TRUE</code> . Must be smaller than or equal to the size of the active risk set. Ignored when <code>sampling = FALSE</code> . Only supported for a tie model.                                                                        |

|             |                                                                                                                                                        |
|-------------|--------------------------------------------------------------------------------------------------------------------------------------------------------|
| seed        | Optional integer. Random seed used for dyad sampling. Setting this ensures reproducible sampling across calls. If NULL, the current RNG state is used. |
| attr_actors | optionally, an object of class "data.frame" that contains exogenous attributes for actors (see Details).                                               |
| attr_dyads  | optionally, an object of class data.frame or matrix containing attribute information for dyads (see Details).                                          |

---

|             |             |
|-------------|-------------|
| remestimate | remestimate |
|-------------|-------------|

---

**Description**

See [remestimate](#).

**Usage**

```
remestimate(  
  reh,  
  stats,  
  approach = c("frequentist", "Bayesian"),  
  random = NULL,  
  penalty = NULL,  
  mixture = NULL,  
  engine = "auto",  
  bayes = list(),  
  seed = NULL,  
  ncores = 1L,  
  WAIC = FALSE,  
  method = NULL,  
  ...  
)
```

**Arguments**

|          |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         |
|----------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| reh      | A remify or remify_durem object.                                                                                                                                                                                                                                                                                                                                                                                                                                                        |
| stats    | A tomstats, tomstats_sampled, aomstats, or remstats_durem object.                                                                                                                                                                                                                                                                                                                                                                                                                       |
| approach | "frequentist" (default) or "Bayesian".                                                                                                                                                                                                                                                                                                                                                                                                                                                  |
| random   | One-sided formula for random effects, e.g. $\sim (1   actor1) + (1   actor2)$ . Fits a GLMM (lme4/glmmTMB, or coxme for ordinal models).                                                                                                                                                                                                                                                                                                                                                |
| penalty  | List of penalisation settings. Frequentist uses glmnet; Bayesian uses shrinkem. Recognised elements: alpha (elastic-net mixing, 1 = lasso (default), 0 = ridge), nfolds (CV folds, default 10), lambda_select ("1se" (default) or "min"), unpenalized / penalized (see below), and prior (shrinkem prior, default "horseshoe"). By default the intercept / baseline structure is left unpenalised: any statistic that is an indicator (all values in {0,1}) is exempt, which covers the |

|         |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               |
|---------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|         | overall baseline, the duration start/end process intercepts (baseline.start / baseline.end) and fixed-effect type dummies (e.g. FEtype_*); count and continuous effect statistics are penalised. Adjust this default with two additive controls (both character vectors of statistic names): unpenalized <i>adds</i> names to the exemption, and penalized <i>removes</i> names from it, i.e. forces them back into the penalty even though they are 0/1 indicators (e.g. a p-shift dummy such as psABAB.end that is a genuine effect, not an intercept). penalized takes precedence when a name is given in both. Names must match the model statistics exactly (as printed by the remstats object, e.g. "psABAB.end", not "psABAB"); a name that matches nothing is ignored with a warning. |
| mixture | List of finite-mixture settings (flexmix). Recognised elements: k (components, default 2), random (clustering formula, e.g. ~ (1   dyad)), concomitant (optional concomitant formula), nrep (random restarts, default 3). E.g. list(k = 2, random = ~ (1   dyad)) fits the dyadic latent class model (Lakdawala et al., 2026).                                                                                                                                                                                                                                                                                                                                                                                                                                                                |
| engine  | GLMM backend: "glmmTMB" or "lme4"; ordinal models automatically use coxme. The default "auto" selects "glmmTMB" when installed (more robust on the stacked tie-oriented design), otherwise falls back to "lme4".                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              |
| bayes   | List of Bayesian (C++ HMC) controls for basic tie/actor models. Recognised elements: nsim (post-burnin iterations, default 2000), nchains (default 2), burnin (default 1000), thin (default 1), init (initial values, default MLE estimates), L (leapfrog steps, default 50), epsilon (leapfrog step size, default 0.002), prior (list with mean and vcov), and nsimWAIC (WAIC draws, default 100).                                                                                                                                                                                                                                                                                                                                                                                           |
| seed    | Random seed.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  |
| ncores  | Number of threads (C++ backends). Default 1L.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 |
| WAIC    | Compute WAIC? Default FALSE.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  |
| method  | [Deprecated] Legacy argument for backward compatibility. Use approach instead. Accepts "MLE" or "HMC" for basic models.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       |
| ...     | Further arguments passed to the backend. The former top-level knobs (alpha, nfolds, lambda_select, k, concomitant, nrep, nsim, nchains, burnin, thin, init, L, epsilon, prior, nsimWAIC) are <i>deprecated</i> but still accepted here and routed into penalty / mixture / bayes.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             |

---

|            |                   |
|------------|-------------------|
| remtribute | <i>remtribute</i> |
|------------|-------------------|

---

**Description**

See [remtribute](#).

**Usage**

```
remtribute(  
  reh,  
  stats = NULL,
```

```
effects = NULL,
attribute = "type",
attribute_type = c("nominal", "ordinal", "numeric"),
attr_actors = NULL,
memory = "full",
memory_value = Inf,
...
)
```

Arguments

|                |                                                                                                                                                                                                                                                                                            |
|----------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| reh            | A remify object whose \$edgelist contains the attribute column (carried through via event_type or event_attributes in <a href="#">remify</a> ). Accepted for both model = "tie" and model = "actor".                                                                                       |
| stats          | A tomstats object (M x D x p), or NULL. When provided, statistics are subsetting internally to the observed dyad at each event. Ignored when effects is supplied.                                                                                                                          |
| effects        | A one-sided formula with tie-model effects, e.g. ~ inertia() + reciprocity(). When provided, tomstats is called internally to compute the statistics. This is the recommended interface when reh uses an actor-oriented model, because it avoids the need for a separate tie-oriented reh. |
| attribute      | Character string: column name in reh\$edgelist that contains the event attribute to model. Default "type".                                                                                                                                                                                 |
| attribute_type | The type of the attribute: "nominal" (unordered categories, multinomial logit), "ordinal" (ordered categories, cumulative link model), or "numeric" (continuous, linear regression).                                                                                                       |
| attr_actors    | Optional data frame of actor-level covariates, passed to tomstats when effects is used.                                                                                                                                                                                                    |
| memory         | Memory type for statistics computation. Passed to tomstats when effects is used. Default "full".                                                                                                                                                                                           |
| memory_value   | Memory parameter value. Passed to tomstats when effects is used.                                                                                                                                                                                                                           |
| ...            | Additional arguments passed to the fitting backend (nnet::multinom, MASS::polr, or stats::glm).                                                                                                                                                                                            |

---

|           |                  |
|-----------|------------------|
| remwindow | <i>remwindow</i> |
|-----------|------------------|

---

Description

See [remwindow](#).

**Usage**

```
remwindow(
  reh,
  stats,
  n.windows = 5L,
  window.width = NULL,
  step.size.window = NULL,
  start.point = 1L,
  min.events = 50L,
  approach = c("frequentist", "Bayesian"),
  parallel = FALSE,
  ncores.window = 1L,
  ...
)
```

**Arguments**

|                               |                                                                                                        |
|-------------------------------|--------------------------------------------------------------------------------------------------------|
| <code>reh</code>              | A remify object.                                                                                       |
| <code>stats</code>            | A tomstats, tomstats_sampled, or aomstats object built on the full reh.                                |
| <code>n.windows</code>        | Target number of windows in auto mode. Default 5.                                                      |
| <code>window.width</code>     | Fixed number of events per window (manual mode).                                                       |
| <code>step.size.window</code> | Events to advance between windows (manual mode only). Default equals <code>window.width</code> .       |
| <code>start.point</code>      | First array row to start windowing from. Default 1L.                                                   |
| <code>min.events</code>       | Floor on window width (auto and manual).                                                               |
| <code>approach</code>         | "frequentist" (default) or "Bayesian", forwarded to <code>remstimate</code> .                          |
| <code>parallel</code>         | Fit windows in parallel? Default FALSE.                                                                |
| <code>ncores.window</code>    | Parallel workers across windows (Unix/macOS only; falls back to sequential with a message on Windows). |
| <code>...</code>              | Further arguments passed to <code>remstimate</code> for every window.                                  |

---

rrankReceive

*rrankReceive*


---

**Description**

See [rrankReceive](#).

**Usage**

```
rrankReceive(consider_type = "ignore")
```

Arguments

`consider_type` character. Controls how event types are handled: "ignore" (default): aggregate over all event types (one statistic); "separate": compute C type-specific statistics, where the type-c statistic for a dyad reflects past type-c events on that actor pair ; "interact": compute C^2 statistics capturing past-event-type x dyad-type interactions (only meaningful with `extend_riskset_by_type=TRUE` in `remify` object). Also accepts FALSE (-> "ignore") and TRUE (-> "separate") for backward compatibility.

---

|                        |                  |
|------------------------|------------------|
| <code>rrankSend</code> | <i>rrankSend</i> |
|------------------------|------------------|

---

Description

See [rrankSend](#).

Usage

```
rrankSend(consider_type = "ignore")
```

Arguments

`consider_type` character. Controls how event types are handled: "ignore" (default): aggregate over all event types (one statistic); "separate": compute C type-specific statistics, where the type-c statistic for a dyad reflects past type-c events on that actor pair ; "interact": compute C^2 statistics capturing past-event-type x dyad-type interactions (only meaningful with `extend_riskset_by_type=TRUE` in `remify` object). Also accepts FALSE (-> "ignore") and TRUE (-> "separate") for backward compatibility.

---

|                   |             |
|-------------------|-------------|
| <code>same</code> | <i>same</i> |
|-------------------|-------------|

---

Description

See [same](#).

Usage

```
same(variable, attr_actors = NULL, attr_data)
```

Arguments

`variable` string with the name of the column in the `attr_actors` object for which the statistic has to be computed.

`attr_actors` optionally, an object of class [data.frame](#) that contains the attribute, see 'Details.'

`attr_data` Deprecated argument. Please use 'attr\_actors' instead.

---

|              |                     |
|--------------|---------------------|
| select_stats | <i>select_stats</i> |
|--------------|---------------------|

---

**Description**

See [select\\_stats](#).

**Usage**

```
select_stats(  
  object,  
  tie_effects = NULL,  
  sender_effects = NULL,  
  receiver_effects = NULL,  
  start_effects = NULL,  
  end_effects = NULL  
)
```

**Arguments**

- object            a remstats object (of class tomstats, aomstats, or remstats\_durem)
- tie\_effects      character vector of statistic names to keep (tomstats only)
- sender\_effects   character vector of statistic names to keep (aomstats only)
- receiver\_effects            character vector of statistic names to keep (aomstats only)
- start\_effects    character vector of statistic names to keep (remstats\_durem only)
- end\_effects      character vector of statistic names to keep (remstats\_durem only)

---

|      |             |
|------|-------------|
| send | <i>send</i> |
|------|-------------|

---

**Description**

See [send](#).

**Usage**

```
send(variable, attr_actors = NULL, scaling = c("none", "std"), attr_data)
```

**Arguments**

|             |                                                                                                                                                                          |
|-------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| variable    | string with the name of the column in the attr_actors object for which the statistic has to be computed.                                                                 |
| attr_actors | optionally, an object of class <code>data.frame</code> that contains the attribute, see 'Details.'                                                                       |
| scaling     | the method for scaling the statistic. Default is to not scale the statistic. Alternatively, standardization of the statistic per time point can be requested with "std". |
| attr_data   | Deprecated argument. Please use 'attr_actors' instead.                                                                                                                   |

sp

*sp***Description**See [sp](#).**Usage**

```
sp(unique = FALSE, scaling = c("none", "std"), consider_type = "ignore")
```

**Arguments**

|               |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          |
|---------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| unique        | A logical value indicating whether to sum the minimum of events with third actors (FALSE, default) or the number of third actors that create a new, unique shared partner (TRUE). See details for more information.                                                                                                                                                                                                                                                                                                                      |
| scaling       | the method for scaling the triad statistic. Default is to not scale the statistic but keep the raw 'counts'. Alternatively, standardization of the raw counts per time point can be requested with 'std'.                                                                                                                                                                                                                                                                                                                                |
| consider_type | character. Controls how event types are handled: "ignore" (default): aggregate over all event types (one statistic); "separate": compute C type-specific statistics, where the type-c statistic for a dyad reflects past type-c events on that actor pair ; "interact": compute $C^2$ statistics capturing past-event-type x dyad-type interactions (only meaningful with <code>extend_riskset_by_type=TRUE</code> in <code>remify</code> object). Also accepts FALSE (-> "ignore") and TRUE (-> "separate") for backward compatibility. |

spUnique

*spUnique***Description**See [spUnique](#).**Usage**

```
spUnique()
```

---

|             |                    |
|-------------|--------------------|
| stack_stats | <i>stack_stats</i> |
|-------------|--------------------|

---

**Description**

See [stack\\_stats](#).

**Usage**

```
stack_stats(stats, reh, add_actors = TRUE)
```

**Arguments**

- |            |                                                                                                                                                                                                                                                                                                                                 |
|------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| stats      | A tomstats object (output of remstats() or tomstats()).                                                                                                                                                                                                                                                                         |
| reh        | A remify object (output of remify::remify()).                                                                                                                                                                                                                                                                                   |
| add_actors | Logical (default TRUE). When TRUE, two extra columns actor1 (sender label) and actor2 (receiver label) are appended by looking up reh\$index\$dyad_map_active (or reh\$riskset_info\$included as a fallback). Set to FALSE to suppress this lookup, e.g. when the riskset has not yet been resolved or for performance reasons. |

---

|     |            |
|-----|------------|
| tie | <i>tie</i> |
|-----|------------|

---

**Description**

See [tie](#).

**Usage**

```
tie(variable, attr_dyads = NULL, scaling = c("none", "std"), x, variableName)
```

**Arguments**

- |            |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |
|------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| variable   | A string specifying the attribute to compute the statistic. If attr_dyads is a data.frame, this refers to the column name in attr_actors. If attr_dyads is a matrix, this corresponds to the name of the exogenous attribute, used to label the statistic in the resulting remstats object.                                                                                                                                                                                                                                    |
| attr_dyads | A data.frame or matrix containing attribute information for dyads. If attr_dyads is a data.frame, the first two columns should represent "actor1" and "actor2" (for directed events, "actor1" corresponds to the sender, and "actor2" corresponds to the receiver). Additional columns can represent dyads' exogenous attributes. If attributes vary over time, include a column named "time". If attr_dyads is a matrix, the rows correspond to "actor1", columns to "actor2", and cells contain dyads' exogenous attributes. |

|              |                                                                                                                                                              |
|--------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------|
| scaling      | The method for scaling the statistic. The default is no scaling. Alternatively, standardization of the statistic per time point can be requested with "std". |
| x            | Deprecated argument. Please use 'attr_dyads' instead.                                                                                                        |
| variableName | Deprecated argument. Please use 'variable' instead.                                                                                                          |

---

|          |                 |
|----------|-----------------|
| tie_data | <i>tie_data</i> |
|----------|-----------------|

---

**Description**

See [tie\\_data](#).

---

|             |                    |
|-------------|--------------------|
| tie_effects | <i>tie_effects</i> |
|-------------|--------------------|

---

**Description**

See [tie\\_effects](#).

**Usage**

```
tie_effects(directed = NULL, endogenous = NULL)
```

**Arguments**

|            |                                                                                                                                                                                       |
|------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| directed   | logical value. The function outputs all statistics in the tie-oriented model for directed events if true, or all statistics in the tie-oriented model for undirected events if false. |
| endogenous | logical value. The function outputs all endogenous statistics in the tie-oriented model if true, or all exogenous statistics if false                                                 |

tomstats

*tomstats***Description**

See [tomstats](#).

**Usage**

```
tomstats(
  tie_effects,
  reh,
  memory = c("full", "window", "decay", "interval"),
  memory_value = NA,
  first = 2,
  last = Inf,
  display_progress = FALSE,
  sampling = FALSE,
  samp_num = 10L,
  seed = NULL,
  attr_actors = NULL,
  attr_dyads = NULL
)
```

**Arguments**

- |                  |                                                                                                                                                                                                                                                                                           |
|------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| tie_effects      | an object of class " <a href="#">formula</a> " (or one that can be coerced to that class): a symbolic description of the tie_effects in the model for which statistics are computed, see 'Details' for the available effects and their corresponding statistics                           |
| reh              | an object of class " <a href="#">remify</a> " characterizing the relational event history. May also be a remify_durem object for duration relational event models.                                                                                                                        |
| memory           | The memory to be used. See 'Details'.                                                                                                                                                                                                                                                     |
| memory_value     | Numeric value indicating the memory parameter. Default is NA, which is only valid for memory = "full" (no memory parameter required). See 'Details'.                                                                                                                                      |
| first            | an optional integer value, specifying the index of the first unique time point event in the relational event history for which statistics must be computed (see 'Details'). Default is 2: the first event has no history and is used only to initialize statistics, not to fit the model. |
| last             | an optional integer value, specifying the index of the last unique time point in the relational event history for which statistics must be computed (see 'Details')                                                                                                                       |
| display_progress | should a progress bar for the computation of the endogenous statistics be shown (TRUE) or not (FALSE)?                                                                                                                                                                                    |
| sampling         | Logical. If TRUE, statistics are computed using case-control (dyad) sampling rather than the full risk set. Default FALSE.                                                                                                                                                                |

|             |                                                                                                                                                                                                  |
|-------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| samp_num    | Integer. Number of dyads to include per event when <code>sampling = TRUE</code> . Must be smaller than or equal to the size of the active risk set. Ignored when <code>sampling = FALSE</code> . |
| seed        | Optional integer. Random seed used for dyad sampling. Setting this ensures reproducible sampling across calls. If <code>NULL</code> , the current RNG state is used.                             |
| attr_actors | optionally, an object of class <code>"data.frame"</code> that contains exogenous attributes for actors (see Details).                                                                            |
| attr_dyads  | optionally, an object of class <code>data.frame</code> or <code>matrix</code> containing attribute information for dyads (see Details).                                                          |

---

|                              |                        |
|------------------------------|------------------------|
| <code>totaldegreeDyad</code> | <i>totaldegreeDyad</i> |
|------------------------------|------------------------|

---

**Description**

See [totaldegreeDyad](#).

**Usage**

```
totaldegreeDyad(scaling = c("none", "prop", "std"), consider_type = "ignore")
```

**Arguments**

- `scaling` the method for scaling the degree statistic. Default is to not scale the statistic (`scaling = "none"`). Alternatively, scaling of the raw degree counts by two times the number of past events at time `t` can be requested with `'prop'` or standardization of the raw degree counts per time point can be requested with `'std'`.
- `consider_type` character. Controls how event types are handled: `"ignore"` (default): aggregate over all event types (one statistic); `"separate"`: compute `C` type-specific statistics, where the type-`c` statistic for a dyad reflects past type-`c` events on that actor pair ; `"interact"`: compute  $C^2$  statistics capturing past-event-type x dyad-type interactions (only meaningful with `extend_riskset_by_type=TRUE` in `remify` object). Also accepts `FALSE` (-> `"ignore"`) and `TRUE` (-> `"separate"`) for backward compatibility.

---

|                                  |                            |
|----------------------------------|----------------------------|
| <code>totaldegreeReceiver</code> | <i>totaldegreeReceiver</i> |
|----------------------------------|----------------------------|

---

**Description**

See [totaldegreeReceiver](#).

**Usage**

```
totaldegreeReceiver(
  scaling = c("none", "prop", "std"),
  consider_type = "ignore"
)
```

**Arguments**

- |               |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              |
|---------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| scaling       | the method for scaling the degree statistic. Default is to not scale the statistic (scaling = "none"). Alternatively, scaling of the raw degree counts by two times the number of past events at time t can be requested with 'prop' or standardization of the raw degree counts per time point can be requested with 'std'.                                                                                                                                                                                 |
| consider_type | character. Controls how event types are handled: "ignore" (default): aggregate over all event types (one statistic); "separate": compute C type-specific statistics, where the type-c statistic for a dyad reflects past type-c events on that actor pair ; "interact": compute C^2 statistics capturing past-event-type x dyad-type interactions (only meaningful with extend_riskset_by_type=TRUE in remify object). Also accepts FALSE (-> "ignore") and TRUE (-> "separate") for backward compatibility. |

---

|                   |                          |
|-------------------|--------------------------|
| totaldegreeSender | <i>totaldegreeSender</i> |
|-------------------|--------------------------|

---

**Description**

See [totaldegreeSender](#).

**Usage**

```
totaldegreeSender(scaling = c("none", "prop", "std"), consider_type = "ignore")
```

**Arguments**

- |               |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              |
|---------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| scaling       | the method for scaling the degree statistic. Default is to not scale the statistic (scaling = "none"). Alternatively, scaling of the raw degree counts by two times the number of past events at time t can be requested with 'prop' or standardization of the raw degree counts per time point can be requested with 'std'.                                                                                                                                                                                 |
| consider_type | character. Controls how event types are handled: "ignore" (default): aggregate over all event types (one statistic); "separate": compute C type-specific statistics, where the type-c statistic for a dyad reflects past type-c events on that actor pair ; "interact": compute C^2 statistics capturing past-event-type x dyad-type interactions (only meaningful with extend_riskset_by_type=TRUE in remify object). Also accepts FALSE (-> "ignore") and TRUE (-> "separate") for backward compatibility. |

---

|          |                 |
|----------|-----------------|
| userStat | <i>userStat</i> |
|----------|-----------------|

---

**Description**

See [userStat](#).

**Usage**

```
userStat(x, variableName = NULL)
```

**Arguments**

- |              |                                                                                                                                                                                                               |
|--------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| x            | Matrix with number of rows equal to the number of events and number of columns equal to the number of dyads in the network (tie-oriented model) or the number of actors in the network (actor-oriented model) |
| variableName | Optionally, a string with the name of the statistic.                                                                                                                                                          |

---

|      |             |
|------|-------------|
| WAIC | <i>WAIC</i> |
|------|-------------|

---

**Description**

See [WAIC](#).

**Usage**

```
WAIC(object, ...)
```

**Arguments**

- |        |                                                      |
|--------|------------------------------------------------------|
| object | is a remstimate object.                              |
| ...    | further arguments to be passed to the 'WAIC' method. |

# Index

- \* **datasets**
  - edgelist\_duration, 22
  - history\_durem, 23
  - randomREH3, 36
- \* **dataset**
  - info3, 26
- \* **data**
  - ao\_data, 16
  - randomREH, 36
  - tie\_data, 56
- activeDegreeDyad, 4, 4
- activeDegreeMax, 4, 4
- activeDegreeMin, 5, 5
- activeIndegreeReceiver, 6, 6
- activeOutdegreeSender, 6, 6
- activeReciprocalTie, 7, 7
- activeSharedPartners, 8, 8
- activeSharedPartners\_isp, 8, 8
- activeSharedPartners\_itp, 9, 9
- activeSharedPartners\_osp, 10, 10
- activeSharedPartners\_otp, 10, 10
- activeTie, 11, 11
- activeTotaldegreeDyad, 12, 12
- activeTotaldegreeReceiver, 12, 12
- activeTotaldegreeSender, 13, 13
- actor\_effects, 14, 14
- AICC, 14, 14
- ao\_data, 16, 16
- aomstats, 15, 15
- average, 16, 16
- bic\_table, 17, 17
- bind\_remstats, 17, 17
- data.frame, 16, 20, 28, 29, 37, 42, 43, 48, 52, 54, 58
- degreeDiff, 18, 18
- degreeMax, 18, 18
- degreeMin, 19, 19
- diagnostics, 19, 19
- difference, 20, 20
- dlcrem, 21, 21
- dyad, 21, 21
- edgelist\_duration, 22
- event, 22, 22
- Ftype, 23, 23
- formula, 15, 47, 57
- frailty\_rem, 23, 23
- history\_durem, 23
- indegreeReceiver, 24, 24
- indegreeSender, 25, 25
- inertia, 25, 25
- info3, 26, 36, 37
- is.remify\_durem, 26, 26
- is.remstats\_durem, 27, 27
- isp, 27, 27
- itp, 28, 28
- maximum, 28, 28
- minimum, 29, 29
- osp, 29, 29
- otp, 30, 30
- outdegreeReceiver, 31, 31
- outdegreeSender, 31, 31
- psABA, 32, 32
- psABAB, 32, 32
- psABAY, 33, 33
- psABB, 33, 33
- psABBA, 33, 33
- psABBY, 34, 34
- psABX, 34, 34
- psABXA, 35, 35
- psABXB, 35, 35
- psABXY, 36, 36

randomREH, [36, 36](#)  
randomREH3, [26, 36](#)  
receive, [37, 37](#)  
recencyContinue, [38, 38](#)  
recencyReceiveReceiver, [38, 38](#)  
recencyReceiveSender, [39, 39](#)  
recencySendReceiver, [39, 39](#)  
recencySendSender, [40, 40](#)  
reciprocity, [40, 40](#)  
remfrailty, [41, 41](#)  
remify, [15, 41, 41, 47, 50, 57](#)  
remixture, [45, 45](#)  
rempenalty, [45, 45](#)  
remstats, [17, 46, 46](#)  
remstimate, [21, 23, 41, 45, 46, 48, 48](#)  
remtribute, [49, 49](#)  
remwindow, [50, 50](#)  
rrankReceive, [51, 51](#)  
rrankSend, [52, 52](#)  
  
same, [52, 52](#)  
select\_stats, [53, 53](#)  
send, [53, 53](#)  
sp, [54, 54](#)  
spUnique, [54, 54](#)  
stack\_stats, [55, 55](#)  
  
tie, [55, 55](#)  
tie\_data, [56, 56](#)  
tie\_effects, [56, 56](#)  
tomstats, [57, 57](#)  
totaldegreeDyad, [58, 58](#)  
totaldegreeReceiver, [58, 58](#)  
totaldegreeSender, [59, 59](#)  
  
userStat, [60, 60](#)  
  
WAIC, [60, 60](#)