Package 'tuple'

July 22, 2025

Type Package

Title Find every match, or orphan, duplicate, triplicate, or other replicated values

Author Emmanuel Lazaridis [aut, cre]

Maintainer Emmanuel Lazaridis <emmanuel@lazaridis.eu>

Depends R (>= 2.10.0)

Description Functions to find all matches or non-matches, orphans, and duplicate or other replicated elements.

License LGPL-3

Encoding UTF-8

LazyLoad no

URL http://statistics.lazaridis.eu

Version 0.4-02

Date 2014-10-31

NeedsCompilation no

Repository CRAN

Date/Publication 2014-10-31 06:07:11

Contents

`tuple-package` *The tuple Package*

Description

Find every match, or orphan, duplicate, triplicate, or other replicated values.

This package extends the base R functionality around checking for `unique` and duplicate values in vectors.

Details

Package:	tuple
Type:	Package
Version:	0.4-02
Date:	2014-10-31
Depends:	R (>= 2.10.0)
Encoding:	UTF-8
License:	LGPL-3
LazyLoad:	no
URL:	http://statistics.lazaridis.eu

Functions to find all matches or non-matches, orphans, and duplicate or other replicated elements.

The following changes are documented since the first release of this package on CRAN:

Version	Change	Description
0.3-06	None	Initial release to CRAN.
0.4-01	Added `%!in%`	This function tests for the opposite of the commonly used testing operator `"%in%"` as documented in `match`.
	Added documentation	Added documentation for the package as a whole. Implemented this change log.
	Improved documentation	Cleaned and otherwise improved documentation that is generated by way of the **roxygen2** package for existing functions.
	Added `tuplicated`	This function is a major addition to the package. It provides a generic way to find elements of a vector that are replicated n or more times. Fundamentally it depends only on the code for `duplicated` as in the first version of this package released to CRAN. The implementation of `triplicated` has not been changed in this in this update from version 0.3-06, but it will be changed to call `tuplicated` with `tuple = 3` in a future release.
	Added `tuplicate`	This function is another major addition. It provides

		a generic way to find elements of a vector that are replicated exactly n times. It depends on the code for the newly-released `tuplicated`, and on the code for `orphan` as in the initial package released to CRAN. The implementation of `triplicate` has not changed from version 0.3-06, but it will be changed to call `tuplicate` with `tuple = 3` in a future release.
0.4-02	Added `matchNone`	This function returns a character string, based on the table, that does not appear in the data.

Author(s)

Emmanuel Lazaridis

`duplicate` *Find Duplicate Values*

Description

Finds values that occur exactly twice in a vector.

Usage

```
duplicate(x)
```

Arguments

`x`	A vector.

Details

Returns the duplicated values in the same order that they would be returned in a call to `orphan`. This fundamentally differs from `duplicated`, which returns a logical vector that is `TRUE` when it runs into any but the first occurrence of a value (and is therefore dependent on the direction of testing of the vector).

See Also

`unique` for similar output, and `duplicated` for the underlying calculations

Examples

```
duplicate(c(NA, 1:3, 3, 4:6, 3, NA, 4))
```

matchAll *Match All Values*

Description

Extends the functionality of match to identify all matching values, instead of just the first one.

Usage

```
matchAll(x, table)
```

Arguments

x	A vector.
table	The lookup table as a vector.

Details

Returns an integer vector of the index in table for all the matches. The result is not sorted in numerical index order when more than one value is sought to be matched. Instead, the matches of the first value in x are listed first, followed by matches to the second value in x and so on. Values of NA are treated as data.

See Also

match

Examples

```
matchAll(3, c(1:3, 3, 4:6, 3, NA, 4))
matchAll(3:4, c(1:3, 3, 4:6, 3, NA, 4))
matchAll(c(NA, 3:4), c(NA, 1:3, 3, 4:6, 3, NA, 4))
```

matchNone *Return a Symbol That Matches No Values*

Description

The tag value is chosen from among special characters so that it does not appear anywhere in the reference input data. The shortest possible tag is chosen.

Usage

```
matchNone(x, table = list(c(".", "!", "/"), c("NA", "na")))
```

Arguments

x	A vector or matrix.
table	The lookup table against which to seek non-matches. This can be a simple vector, or it can be a list of two vectors.

Details

This function is used in other packages by the same author to extend missing data handling in R. It provides for flexible missing data identifiers where needed by an S4 class, and similar unmatched identifiers for other dirty data problems.

Value

A string composed of the strings in the `table`. The default list choses the first non-matching value out of 179 values that are unlikely to be used in most real sets of data. If only `table` is specified, the possible values for a non-matching string, ordered from the most to the least preferable, are returned.

Examples

```
my.x <- c(1,2,3,2,3,1,2)
matchNone(my.x)
matchNone(c(my.x,"."))
matchNone(c(my.x,".","!"))
matchNone(c(my.x,".","!","/"))
matchNone(c(my.x,".","!","/",".."))
matchNone(table = ".")
```

---

not-in *Mismatch Test*

Description

Test whether some data are not in a table.

Usage

```
x %!in% table
```

Arguments

x	A vector of data.
table	A table of reference values.

Details

This helps avoid code structures like `!(x %in% table)`.

See Also

match

Examples

```
1:2 %!in% 2:4
```

---

orphan *Find Orphan Values*

Description

Finds values that occur exactly once in a vector.

Usage

```
orphan(x)
```

Arguments

x	A vector.

Details

Returns the unique values in the same order that they would be returned in a call to unique.

See Also

unique

Examples

```
orphan(c(NA, 1:3, 3, 4:6, 3, NA, 4))
```

triplicate *Find Triplicate Values*

Description

Finds values that occur exactly three times in a vector.

Usage

```
triplicate(x)
```

Arguments

x	A vector.

Details

Returns the triplicated values in the same order that they would be returned in a call to orphan. This fundamentally differs from triplicated, which returns a logical vector that is TRUE when it runs into any but the first or second occurrences of a value (and is therefore dependent on the direction of testing of the vector).

See Also

duplicate

Examples

```
triplicate(c(NA, 1:3, 3, 4:6, 3, NA, 4))
triplicate(c(NA, 1:3, 3, 4:6, 3, NA, 4, 3))
```

triplicated *Find Values That Are Repeated At Least Thrice*

Description

Finds values that are repeated at least three times in a vector.

Usage

```
triplicated(x, ..., fromLast = FALSE)
```

Arguments

x	A vector.
...	Other optional arguments are ignored.
fromLast	A logical indicating if triplication should be considered from the reverse side, i.e., the two last (or rightmost) of identical elements would return FALSE.

Details

Returns a logical vector that is TRUE when it runs into any but the first or second occurrences of a value, analogous to duplicated.

See Also

duplicated

Examples

```
triplicated(c(NA, 1:3, 3, 4:6, 3, NA, 4, 3))
```

---

tuplicate *Find n-Replicated Elements*

Description

Finds elements that occur exactly n times in a vector.

Usage

```
tuplicate(x, n)
```

Arguments

x	A vector.
n	An integer.

Details

Returns the n-replicated elements in the same order that they would be returned in a call to orphan. This fundamentally differs from tuplicated, which returns a logical vector that is TRUE when it runs into any but the (n-1)-st and fewer occurrences of an element (and is therefore dependent on the direction of testing of the vector).

See Also

duplicate

Examples

```
x <- c(NA, 1:3, 4:5, rep(6, 6), 3, NA, 4, 3, 3)
lapply(2:6, function(X) { tuplicate(x, X) })
```

`tuplicated` *Find Elements That Are Repeated At Least n Times*

Description

Finds elements that are repeated at least n times in a vector.

Usage

```
tuplicated(x, n, ..., fromLast = FALSE)
```

Arguments

`x`	A vector.
`n`	An integer.
`...`	Other optional arguments are ignored.
`fromLast`	A logical indicating if n-replication should be considered from the right side of the vector. If `TRUE`, the `n-1` last (or rightmost) of replicated identical elements return `FALSE`.

Details

Returns a logical vector that is `TRUE` when it runs into any but the `(n-1)`-st occurrences of an element, analogous to `duplicated`.

See Also

`duplicated`

Examples

```
x <- c(NA, 1:3, 4:5, rep(6, 6), 3, NA, 4, 3, 3)
all(tuplicated(x, 3) == triplicated(x))
```

Index