

Package ‘mulset’

May 9, 2026

Version 1.0.0

Title Multiset Intersection Generator

Date 2019-03-7

Author Ivan Tomic [aut, cre, cph] (ORCID:
<<https://orcid.org/0000-0003-3596-681X>>),
Adriana Tomic [aut, ctb] (ORCID:
<<https://orcid.org/0000-0001-9885-3535>>)

Description

Computes efficient data distributions from highly inconsistent datasets with many missing values using multi-set intersections. Based upon hash functions, 'mulset' can quickly identify intersections from very large matrices of input vectors across columns and rows and thus provides scalable solution for dealing with missing values. Tomic et al. (2019) <[doi:10.1101/545186](https://doi.org/10.1101/545186)>.

Maintainer Ivan Tomic <info@ivantomic.com>

Imports gtools, digest, stats

Depends R (>= 3.4.0)

URL <https://github.com/LogIN-/mulset>

BugReports <https://github.com/LogIN-/mulset/issues>

License EUPL (>= 1.2)

Encoding UTF-8

LazyLoad yes

LazyData yes

RoxygenNote 6.1.1.9000

NeedsCompilation no

Repository CRAN

Date/Publication 2019-03-08 16:50:03 UTC

Contents

intersection	2
mulset	2
mulsetDemo	3

intersection	<i>A intersection function</i>
--------------	--------------------------------

Description

intersection() returns all intersections it found.

Usage

```
intersection(...)
```

Arguments

... Vector with master values to check and vector to compare values against

Value

Character vector of all common attributes

Examples

```
input1 <- seq(50, 100, by=10)
input2 <- seq(70, 130, by=10)
intersection(input1, input2)
```

mulset	<i>A mulset function</i>
--------	--------------------------

Description

mulset() returns all multi-set intersections

Usage

```
mulset(data, exclude = NULL, include = c("samples", "samples_count",
    "datapoints"), maxIntersections = NULL, hashMethod = "md5")
```

Arguments

data	Data frame containing your data
exclude	Vector containing one or more variable names from names(data)
include	List of attributes which will be shown in results. Possible values are: c("samples", "samples_count", "datapoints"). If parameter is set to NULL only c("features", "feature_count") will be returned.
maxIntersections	Maximum number of unique datasets to generate, if NULL all datasets will be generated
hashMethod	Hashing method to use for unique sets identification. Available choices: md5(default), sha1, crc32, sha256, sha512, xxhash32, xxhash64, murmur32

Details

This function allows you to generate specific type of multi-set intersections. It searches for multi set intersections between rows and column identifiers. If no NA values are present only 1 dataset is returned as expected.

Value

If any intersections are found it returns a list that contains all available multi-set intersections You can convert this to data-frame following example provided or use it as it is.

Examples

```
data(mulsetDemo)
print(head(mulsetDemo))
resamples <- mulset(mulsetDemo, exclude = c("outcome", "age", "gender"), maxIntersections = 250)
## Loop through returned list or convert it to data-frame
## resamplesFrame <- as.data.frame(t(sapply(resamples,c)))
```

mulsetDemo	<i>Demo data set from mulset package. This data is used in this package examples. It consist of 4x4 feature matrix + additional dummy columns that can be used for testing.</i>
------------	---

Description

Demo data set from mulset package. This data is used in this package examples. It consist of 4x4 feature matrix + additional dummy columns that can be used for testing.

Usage

```
data(mulsetDemo)
```

Format

An object of class `data.frame` with 4 rows and 7 columns.

Examples

```
data(mulsetDemo)
print(head(mulsetDemo))
resamples <- mulset(mulsetDemo, exclude = c("outcome", "age", "gender"))
```

Index

- * **datasets**
 - mulsetDemo, 3
 - * **data**
 - mulset, 2
 - * **intersect,**
 - intersection, 2
 - * **intersection,**
 - mulset, 2
 - * **intersection**
 - intersection, 2
 - * **missing**
 - mulset, 2
 - * **mulset,**
 - mulset, 2
 - * **multi-set**
 - mulset, 2
 - * **table**
 - mulset, 2
- intersection, 2
- mulset, 2
- mulsetDemo, 3