Package 'yarr'

October 14, 2022

Type Package

Title Yet Another 'ARFF' Reader

Version 0.1.2

Description A parser and a writer for 'WEKA' Attribute-Relation File Format <https://waikato.github.io/weka-wiki/arff_stable/> in pure R, with no dependencies. As opposed to other R implementations, this package can read standard (dense) as well as sparse files, i.e. those where each row does only contain nonzero components. Unlike 'RWeka', 'yarr' does not require any 'Java' installation nor is dependent on external software. This implementation is generalized from those in packages 'mldr' and 'mldr.datasets'.

License GPL (>= 3)

Encoding UTF-8

LazyData true

URL https://github.com/fdavidcl/yarr

BugReports https://github.com/fdavidcl/yarr/issues

Depends R (>= 3.6)

Suggests testthat (>= 2.1.0)

RoxygenNote 6.1.1

NeedsCompilation no

Author David Charte [aut, cre] (<https://orcid.org/0000-0002-4830-9512>), Francisco Charte [aut] (<https://orcid.org/0000-0002-3083-8942>)

Maintainer David Charte <fdavidcl@ugr.es>

Repository CRAN

Date/Publication 2019-08-10 12:20:02 UTC

R topics documented:

`attr.names` *Dataset utilities*

Description

Some tools to access the metadata in ARFF files. `attr.names` retrieves the names of the attributes, `attr.types` returns the ARFF type of each variable, `relation` shows the name/relation specified in the `@relation` section of the file.

Usage

```
attr.names(x)

attr.types(x)

relation(x)
```

Arguments

`x`	A dataset read using `read.arff`

`print.arff_data` *Display functions*

Description

Display functions

Usage

```
## S3 method for class 'arff_data'
print(x, max_attrs = 10, max_values = 5, ...)
```

Arguments

`x`	A data.frame read from an ARFF file
`max_attrs`	Maximum of attributes to be inspected
`max_values`	Maximum of values to be shown for each attribute
`...`	Extra parameters for the corresponding S3 method for class `data.frame`

read.arff *Read an ARFF file*

Description

Reads a dataset from an ARFF file, parsing each section and converting the data section into a `data.frame`.

Usage

```
read.arff(file, stringsAsFactors = FALSE)
```

Arguments

`file`	Name of the file to read the data from
`stringsAsFactors`	Logical: should string attributes be converted to factors? (nominal attributes are always converted to factors)

Value

A `data.frame` with some attributes:

- attributes: a named vector indicating the type of each variable
- relation: the original `@relation` of the dataset

Use `attr.names()`, `attr.types()` and `relation()` to consult attribute names, types and the name of the dataset, respectively.

Examples

```
library(yarr)

yeast <- read.arff("yeast.arff")
```

write.arff *Write a data.frame onto an ARFF file*

Description

Takes a data frame and records it in ARFF (Attribute-Relation File Format).

Usage

```
write.arff(x, relation = NULL, types = NULL, file = "",
  sparse = FALSE, append = FALSE, ...)
```

Arguments

x	A data.frame
relation	Name of the dataset (optional, it may be inferred from the `relation` attribute or the name of the variable passed as argument)
types	A character vector indicating the type of each variable (optional, may be inferred from the `attributes` attribute or computed from the class of each variable)
file	Name of the file where the data is to be written. Use `""` to write to standard output
sparse	Logical: write in sparse format?
append	Logical: append to an existing file?
...	Extra parameters for internal functions

Value

Invisibly, the name of the file.

Examples

```
library(yarr)

write.arff(iris, "iris", file = tempfile())
```

Index