Package: filematrix (via r-universe)

November 1, 2024

Type Package

Title File-Backed Matrix Class with Convenient Read and Write Access

Version 1.3

Date 2018-02-26

Description Interface for working with large matrices stored in files,
not in computer memory. Supports multiple non-character data
types (double, integer, logical and raw) of various sizes (e.g.
8 and 4 byte real values). Access to parts of the matrix is
done by indexing, exactly as with usual R matrices. Supports
very large matrices. Tested on multi-terabyte matrices. Allows
for more than 2^32 rows or columns. Allows for quick addition
of extra columns to a filematrix. Cross-platform as the package
has R code only.

BugReports https://github.com/andreyshabalin/filematrix/issues

URL https://github.com/andreyshabalin/filematrix

License LGPL-3

Depends methods, utils

VignetteBuilder knitr

Suggests knitr, rmarkdown, RSQLite

Repository https://andreyshabalin.r-universe.dev

RemoteUrl https://github.com/andreyshabalin/filematrix

RemoteRef HEAD

RemoteSha b9b5872b87d731d9f9e1ed5c97651e4337442d94

Contents

---

filematrix-package   *File-backed numeric matrix.*

Description

File-Backed Matrix Class with Convenient Read and Write Access

Details

Interface for working with large matrices stored in files, not in computer memory. Supports multiple non-character data types (double, integer, logical and raw) of various sizes (e.g. 8 and 4 byte real values). Access to parts of the matrix is done by indexing (e.g. fm[,1]), exactly as with usual R matrices. Supports very large matrices. Tested on multi-terabyte matrices. Allows for more than 2^32 rows or columns. Allows for quick addition of extra columns to a filematrix. Cross-platform as the package has R code only.

A new file.matrix object can be created with fm.create and fm.create.from.matrix. Existing file.matrix files can be opened with fm.open.

Once a file.matrix is created or opened it can be accessed as a regular matrix object in R. All changes to file.matrix object are written to the data files without extra buffering.

Note

Due to lack of 64 bit integer data type in R, the package uses double values for calculation of indices. The precision of double data type is sufficient for indexing matrices up to 8,192 terabytes in size.

Author(s)

Andrey A Shabalin <andrey.shabalin@gmail.com>

See Also

See fm.create and filematrix for reference.

Run browseVignettes("filematrix") for the list of vignettes.

---

filematrix-class   *Manipulating file matrices (class* "filematrix"*)*

Description

`filematrix` is a class for working with very large matrices stored in files, not held in computer memory. It is intended as a simple, efficient solution to handling big numeric data (i.e., datasets larger than memory capacity) in R.

A new filematrix can be created with `fm.create`. It can be created from an existing R matrix with `fm.create.from.matrix`. A text file with a matrix can be scanned and converted into a filematrix with `fm.create.from.text.file`. An existing filematrix can be opened for read/write access with `fm.open` or loaded fully in memory with `fm.load`.

A filematrix can be handled as an ordinary matrix in R.

It can be read from and written to via usual indexing with possible omission of indices.
For example: `fm[1:3,2:4]` and `fm[,2:4]`.

The values can also be accessed as a vector with single indexing.
For example: `fm[3:7]` and `fm[4:7] = 1:4`.

A whole filematrix can be read memory as an ordinary R matrix with `as.matrix` function or empty indexing `fm[]`.

The dimensions of filematrix can be obtained via `dim`, `nrow` and `ncol` functions and modified with `dim` function.
For example: `dim(fm)` and `dim(fm) = c(10,100)`.

The number of elements in filematrix is returned by `length` function.

A filematrix can have row and column names. They can be accessed using the standard functions `rownames`, `colnames`, and `dimnames`.

A filematrix can be closed after use with `close` command. Note, however, that there is no risk of losing modifications to a filematrix if an object is not closed, as all changes are written to disk without delay.

Usage

```
## S3 method for class 'filematrix'
x[i,j]
## S3 replacement method for class 'filematrix'
x[i,j] <- value

## S4 method for signature 'filematrix'
as.matrix(x)

## S4 method for signature 'filematrix'
dim(x)
## S4 replacement method for signature 'filematrix'
dim(x) <- value

## S4 method for signature 'filematrix'
length(x)

## S4 method for signature 'filematrix'
rownames(x)
## S4 replacement method for signature 'filematrix'
```

```
rownames(x) <- value

## S4 method for signature 'filematrix'
colnames(x)
## S4 replacement method for signature 'filematrix'
colnames(x) <- value

## S4 method for signature 'filematrix'
dimnames(x)
## S4 replacement method for signature 'filematrix'
dimnames(x) <- value
```

Arguments

x	A filematrix object (`filematrix`).
i, j	Row/column indices specifying elements to extract or replace.
value	A new value to replace the indexed element(s).

Value

`length` function returns the number of elements in the filematrix.

Functions `colnames`, `rownames`, and `dimnames` return the same values as their counterparts for the regular R matrices.

Methods

isOpen Returns `TRUE` is the filematrix is open.

readAll(): Return the whole matrix.
>   Same as `fm[]` or `as.matrix(fm)`

writeAll(value): Fill in the whole matrix.
>   Same as `fm[] = value`

readSubCol(i, j, num): Read num values in column j starting with row i.
>   Same as `fm[i:(i+num-1), j]`

writeSubCol(i, j, value): Write values in the column j starting with row i.
>   Same as `fm[i:(i+length(value)-1), j] = value`

readCols(start, num): Read num columns starting with column start.
>   Same as `fm[, start:(start+num-1)]`

writeCols(start, value): Write columns starting with column start.
>   Same as `fm[, start:(start+ncol(value)-1)] = value`

readSeq(start, len): Read len values from the matrix starting with start-th value.
>   Same as `fm[start:(start+len-1)]`

writeSeq(start, value): Write values in the matrix starting with start-th value.
>   Same as `fm[start:(start+length(value)-1)] = value`

appendColumns(mat) Increases filematrix by adding columns to the right side of the matrix. Matrix mat must have the same number of rows.
>   Same as `fm = cbind(fm, mat)` for ordinary matrices.

Author(s)

Andrey A Shabalin <andrey.shabalin@gmail.com>

See Also

For function creating and opening file matrices see `fm.create`.

Run browseVignettes("filematrix") for the list of vignettes.

| fm.create | *Functions to Create a New, or Open an Existing Filematrix* |

Description

Create a new or open existing `filematrix` object.

`fm.create` creates a new filematrix. If a filematrix with this name exists, it is overwritten (destroyed).

`fm.create.from.matrix` creates a new filematrix copy of an existing R matrix.

`fm.open` opens an existing filematrix for read/write access.

`fm.load` loads entire existing filematrix into memory as an ordinary R matrix.

`fm.create.from.text.file` reads a matrix from a text file into a new filematrix. The rows in the text file become columns in the filematrix. The transposition happens because the text files stores data by rows and filematrices store data by columns.

Usage

```
fm.create(
    filenamebase,
    nrow = 0,
    ncol = 1,
    type = "double",
    size = NULL,
    lockfile = NULL)

fm.create.from.matrix(
    filenamebase,
    mat,
    size = NULL,
    lockfile = NULL)

fm.open(
    filenamebase,
    readonly = FALSE,
    lockfile = NULL)

fm.load(filenamebase, lockfile = NULL)
```

```
fm.create.from.text.file(
    textfilename,
    filenamebase,
    skipRows = 1,
    skipColumns = 1,
    sliceSize = 1000,
    omitCharacters = "NA",
    delimiter = "\t",
    rowNamesColumn = 1,
    type = "double",
    size = NULL)

## S4 method for signature 'filematrix'
close(con)

closeAndDeleteFiles(con)
```

Arguments

filenamebase	Name without extension for the files storing the filematrix. The file `<filenamebase>.bmat` keeps the matrix values and `<filenamebase>.desc.txt` stores the matrix dimensions, data type, and data type size. Names of rows and columns, if defined, are stored in `<filenamebase>.nmsrow.txt` and `<filenamebase>.nmscol.txt`.
nrow	Number of rows in the matrix. Values over 2^32 are supported.
ncol	Number of columns in the matrix. Values over 2^32 are supported.
type	The type of values stored in the matrix. Can be either `"double"`, `"integer"`, `"logical"`, or `"raw"`.
size	Size of each item of the matrix in bytes. Default values are 8 for `"double"`, 4 for `"integer"`, and 1 for `"logical"` and `"raw"`. Do not change if not sure.
mat	Regular R matrix, to be copied into a new filematrix.
readonly	If `TRUE`, the values in the opened filematrix cannot be changed.
textfilename	Name of the text file with matrix data, to be copied into a new filematrix.
skipRows	Number of rows with column names. The matrix values are expected after first `skipRows` rows of the file. Can be zero.
skipColumns	Number of columns before matrix values begin. Can be zero.
sliceSize	The text file with matrix is read in chuncks of `sliceSize` rows. This is a performance tuning parameter, it does not affect the outcome.
omitCharacters	The text string representing missing values. Default value is NA.
delimiter	The delimiter separating values in the text matrix file.
rowNamesColumn	The row names are taken from the `rowNamesColumn`-th column of the text file. By default, row names are extracted from the first column.

con   A filematrix object.

lockfile   Optional. Name of a lock file (file is overwritten). Used to avoid simultaneous
                operations by multiple R instances accessing the same filematrix or different
                filematrices on the same hard drive. Do not use if not sure.

Details

Once created or opened, a filematrix object can be accessed as an ordinary matrix using both matrix
`fm[,]` and vector `fm[]` indexing. The indices can be integer (no zeros) or logical vectors.

Value

Returns a filematrix object. The object can be closed with `close` command or closed and deleted
from disk with closeAndDeleteFiles command.

Author(s)

Andrey A Shabalin <andrey.shabalin@gmail.com>

See Also

For more on the use of filematrices see filematrix.

Run `browseVignettes("filematrix")` for the list of vignettes.

Examples

```
# Create a 10x10 matrix
fm = fm.create(filenamebase=tempfile(), nrow=10, ncol=10)

# Change values in the top 3x3 corner
fm[1:3,1:3] = 1:9

# View the values in the top 4x4 corner
fm[1:4,1:4]

# Close and delete the filematrix
closeAndDeleteFiles(fm)
```

Index