Package 'bigreadr'

October 18, 2019

Version 0.2.0

Date 2019-10-17

Title Read Large Text Files

Description Read large text files by splitting them in smaller files.
Package 'bigreadr' also provides some convenient wrappers around fread()
and fwrite() from package 'data.table'.

License GPL-3

Encoding UTF-8

LazyData true

ByteCompile true

RoxygenNote 6.1.0

Imports bigassertr (>= 0.1.1), data.table, Rcpp, parallel, utils

Suggests spelling, testthat, covr, RSQLite

LinkingTo Rcpp

Language en-US

URL https://github.com/privefl/bigreadr

BugReports https://github.com/privefl/bigreadr/issues

NeedsCompilation yes

Author Florian Privé [aut, cre]

Maintainer Florian Privé <florian.prive.21@gmail.com>

Repository CRAN

Date/Publication 2019-10-18 04:50:08 UTC

R topics documented:

`big_fread1` *Read large text file*

Description

Read large text file by splitting lines.

Usage

```
big_fread1(file, every_nlines, .transform = identity,
  .combine = rbind_df, skip = 0, ..., print_timings = TRUE)
```

Arguments

`file`	Path to file that you want to read.
`every_nlines`	Maximum number of lines in new file parts.
`.transform`	Function to transform each data frame corresponding to each part of the `file`. Default doesn't change anything.
`.combine`	Function to combine results (list of data frames).
`skip`	Number of lines to skip at the beginning of `file`.
`...`	Other arguments to be passed to data.table::fread, excepted `input`, `file`, `skip`, `col.names` and `showProgress`.
`print_timings`	Whether to print timings? Default is `TRUE`.

Value

A `data.frame` by default; a `data.table` when `data.table = TRUE`.

big_fread2 *Read large text file*

Description

Read large text file by splitting columns.

Usage

```
big_fread2(file, nb_parts = NULL, .transform = identity,
  .combine = cbind_df, skip = 0, select = NULL, progress = FALSE,
  part_size = 500 * 1024^2, ...)
```

Arguments

`file`	Path to file that you want to read.
`nb_parts`	Number of parts in which to split reading (and transforming). Parts are referring to blocks of selected columns. Default uses `part_size` to set a good value.
`.transform`	Function to transform each data frame corresponding to each block of selected columns. Default doesn't change anything.
`.combine`	Function to combine results (list of data frames).
`skip`	Number of lines to skip at the beginning of `file`.
`select`	Indices of columns to keep (sorted). Default keeps them all.
`progress`	Show progress? Default is `FALSE`.
`part_size`	Size of the parts if `nb_parts` is not supplied. Default is `500 * 1024^2` (500 MB).
`...`	Other arguments to be passed to data.table::fread, excepted `input`, `file`, `skip`, `select` and `showProgress`.

Value

The outputs of `fread2` + `.transform`, combined with `.combine`.

cbind_df *Merge data frames*

Description

Merge data frames

Usage

```
cbind_df(list_df)
```

Arguments

list_df	A list of multiple data frames with the same observations in the same order.

Value

One merged data frame.

Examples

```
str(iris)
str(cbind_df(list(iris, iris)))
```

---

fread2 *Read text file(s)*

Description

Read text file(s)

Usage

```
fread2(input, ..., data.table = FALSE,
  nThread = getOption("bigreadr.nThread"))
```

Arguments

input	Path to the file(s) that you want to read from. This can also be a command, some text or an URL. If a vector of inputs is provided, resulting data frames are appended.
...	Other arguments to be passed to data.table::fread.
data.table	Whether to return a data.table or just a data.frame? Default is FALSE (and is the opposite of data.table::fread).
nThread	Number of threads to use. Default uses all threads minus one.

Value

A data.frame by default; a data.table when data.table = TRUE.

Examples

```
tmp <- fwrite2(iris)
iris2 <- fread2(tmp)
all.equal(iris2, iris)  ## fread doesn't use factors
```

fwrite2 *Write a data frame to a text file*

Description

Write a data frame to a text file

Usage

```
fwrite2(x, file = tempfile(), ..., quote = FALSE,
  nThread = getOption("bigreadr.nThread"))
```

Arguments

x	Data frame to write.
file	Path to the file that you want to write to. Defaults uses `tempfile()`.
...	Other arguments to be passed to data.table::fwrite.
quote	Whether to quote strings (default is `FALSE`).
nThread	Number of threads to use. Default uses all threads minus one.

Value

Input parameter `file`, invisibly.

Examples

```
tmp <- fwrite2(iris)
iris2 <- fread2(tmp)
all.equal(iris2, iris)  ## fread doesn't use factors
```

nlines *Number of lines*

Description

Get the number of lines of a file.

Usage

```
nlines(file)
```

Arguments

file	Path of the file.

Value

The number of lines as one integer.

Examples

```
tmp <- fwrite2(iris)
nlines(tmp)
```

`rbind_df` *Merge data frames*

Description

Merge data frames

Usage

```
rbind_df(list_df)
```

Arguments

`list_df`	A list of multiple data frames with the same variables in the same order.

Value

One merged data frame with the names of the first input data frame.

Examples

```
str(iris)
str(rbind_df(list(iris, iris)))
```

`split_file` *Split file every nlines*

Description

Split file every nlines

Get files from splitting.

Usage

```
split_file(file, every_nlines, prefix_out = tempfile(),
  repeat_header = FALSE)

get_split_files(split_file_out)
```

Arguments

`file`	Path to file that you want to split.
`every_nlines`	Maximum number of lines in new file parts.
`prefix_out`	Prefix for created files. Default uses `tempfile()`.
`repeat_header`	Whether to repeat the header row in each file. Default is `FALSE`.
`split_file_out`	Output of split_file.

Value

A list with

- `name_in`: input parameter `file`,
- `prefix_out`: input parameter 'prefix_out",
- `nfiles`: Number of files (parts) created,
- `nlines_part`: input parameter `every_nlines`,
- `nlines_all`: total number of lines of `file`.

Vector of file paths created by split_file.

Examples

```
tmp <- fwrite2(iris)
infos <- split_file(tmp, 100)
str(infos)
get_split_files(infos)
```

Index