Package: collUtils (via r-universe)

October 21, 2024

Type Package

Title Auxiliary Package for Package 'CollapsABEL'

Version 1.0.5

Date 2016-03-26

Author Kaiyin Zhong, Fan Liu

Maintainer Kaiyin Zhong <kindlychung@gmail.com>

Depends R (>= 3.1.3), rJava (>= 0.9-6), Rcpp (>= 0.11.2)

LinkingTo Rcpp

Description Provides some low level functions for processing PLINK input and output files.

URL https://bitbucket.org/kindlychung/collutils

BugReports https://bitbucket.org/kindlychung/collutils/issues

Suggests testthat

SystemRequirements Java (>= 1.6)

License GPL-3

NeedsCompilation yes

Repository CRAN

Date/Publication 2016-03-31 19:15:26

Contents

collUtils-package *A auxiliary package for CollapsABEL.*

Description

This package includes some low level functions for processing PLINK input and output files written in Java or C++. Normally you shouldn't need to directly use functions from this package.

Details

Package:	collUtils
Type:	Package
Version:	1.0
Date:	2015-06-12
License:	GPL-3

Author(s)

Kaiyin Zhong Maintainer: Kaiyin Zhong <kindlychung@gmail.com>

References

To be updated.

Examples

```
## Not run:
require(collUtils)
rbed_obj = rBed("test.bed")
geno = rbed_obj$readBed()
geno = getJArray(geno)
print(geno)
fn = tempfile()
f = file(fn, "wb")
writeBin("a", f)
writeBin("b", f)
writeBin("c", f)
close(f)
file.info(fn)$size == 6
truncateEndOfFile(fn, 1)

## End(Not run)
```

countlines *Count the number of lines in a file*

Description

Count the number of lines in a file

Usage

```
countlines(fn)
```

Arguments

fn	Input filepath

Value

A integer for the number of lines

getJArray *Import Java array into R*

Description

A thin wrapper around `rJava::.jevalArray`

Usage

```
getJArray(mat_ref, na_vals = -9)
```

Arguments

mat_ref	Reference object of the Java array
na_vals	NA code. Default to -9.

Author(s)

Kaiyin Zhong

ncols *Counts the number of columns of whitespace delimited file.*

Description

Counts the number of columns of whitespace delimited file.

Usage

```
ncols(fn)
```

Arguments

fn	Input filepath

Value

A integer for the number of columns

rBed *Wrapper for constructor of Bed class*

Description

Wrapper for constructor of Bed class

Usage

```
rBed(bed_path, bytes_snp = NULL, nindiv = NULL)
```

Arguments

bed_path	character. Path to bed file.
bytes_snp	integer. Bytes per SNP.
nindiv	integer. Number of individuals.

Value

jobjRef object.

Author(s)

Kaiyin Zhong

Examples

```
## do not run
# rbed_obj = rBed("test.bed")
# geno = rbed_obj$readBed()
# geno = getJArray(geno)
# print(geno)
```

`readcol` *Read one column of a whitespace delimited text file*

Description

Read one column of a whitespace delimited text file

Usage

```
readcol(fileName, colNum, nSkip, maxRowNum)
```

Arguments

`fileName`	Input filepath
`colNum`	An integer. The target column number
`nSkip`	An integer. Number of lines to skip in the beginning.
`maxRowNum`	An Integer. Maximum number of lines to read

Value

A vector of strings containing the target column

`readcols` *Read columns of a whitespace delimited text file*

Description

Read columns of a whitespace delimited text file

Usage

```
readcols(fn, colsel, nFirstSkipLines, nSkipUnit)
```

Arguments

`fn`	input filepath
`colsel`	a vector of target column numbers
`nFirstSkipLines`	Integer. Number of lines to skip in the beginning
`nSkipUnit`	Integer M. Let the function read one line out of every M

Value

A matrix of strings from selected columns

`truncateEndOfFile` *Truncate n bytes from end of file*

Description

Truncate n bytes from end of file

Usage

```
truncateEndOfFile(filename, len)
```

Arguments

`filename`	character. Filename.
`len`	numeric. Number of bytes to truncate

Author(s)

Kaiyin Zhong

Examples

```
## Not run:
fn = tempfile()
f = file(fn, "wb")
writeBin("a", f)
writeBin("b", f)
writeBin("c", f)
close(f)
file.info(fn)$size == 6
truncateEndOfFile(fn, 1)
file.info(fn)$size == 5

## End(Not run)
```

Index