Package 'RCytoGPS'

September 17, 2021

Version 1.2.1

Date 2021-09-17

Title Using Cytogenetics Data in R

Author Kevin R. Coombes, Dwayne Tally

Maintainer Kevin R. Coombes <krc@silicovore.com>

Description Defines classes and methods to process text-based cytogenetics using the CytoGPS web site, then import the results into R for further analysis and graphing.

Depends R (>= 3.5)

Suggests knitr, rmarkdown, Mercator

Imports methods, graphics, grDevices, rjson

VignetteBuilder knitr

License Apache License (== 2.0)

URL http://oompa.r-forge.r-project.org/

NeedsCompilation no

Repository CRAN

Date/Publication 2021-09-17 14:40:02 UTC

R topics documented:

Chromosome-class *The* Chromosome *Class*

Description

A class to represent a single chromsome in order to plot an image of the Giemsa-stained cytobands.

Usage

```
Chromosome(I, res = 2500, maxbase = 250000000)
## S4 method for signature 'Chromosome'
image(x, horiz = FALSE, mai = NULL, showBandNames = FALSE, ...)
```

Arguments

I	A human chromosome identifier; one of the values in c(1:22,"X","Y").
res	An integer resolution; the number of pixels used to represent an entire chromosome.
maxbase	An integer representing an upper bound on the length of the longest chromosome, mneasured in base-pairs.
x	An object of the Chromosome class.
horiz	A logical value: should the image of the chromsome be oriented horizontally.
mai	margin inches, as in the usual graphical argument to par.
showBandNames	logical; shold the names of the bands be written on the plot?
...	Additional arguments to the image method; ignored.

Details

Conventional karyotyping describes chromosomal abnormalities (in a standardized text-based nomenclature) that are visible through a microsope. Karyotyping relies on a technique known as Giemsa staining, which creates a banding pattern along the chromosome of different shades of gray. This class is used to plot images of individual chromosomes, corectly reflecting the size and color of the bands..

Value

The Chromosome constructor returns an object of the Chromosome class. The image method invisibly returns its first argument.

Slots

name: A character value, typically of the form "chr1".

label: A chartacter value, typiclly of the form "Chr 1".

grid: An integer vector (of length res) repesenting the base position along the chromosome.

`range`: An integer vector of length 2 marking teh starting and ending position of the banded part of the chromosome, in bases.

`stain`: An integer vector (of length `res`), where the values indicate the color of the Giemsa stain for that part of the chromosome.

Methods

image: `signature(object = "RCytoGPS")` Creates an image of the chromosme, with bands colored according to Giemsa staining.

Author(s)

Kevin R. Coombes <krc@silicovore.com>, Dwayne G. Tally <dtally110@hotmail.com>

References

Abrams ZB, Tally DG, Coombes KR. RCytoGPS: An R Package for Analyzing and Visualizing Cytogenetic Data. In preparation.

Abrams ZB, Tally DG, Zhang L, Coombes CE, Payne PRO, Abruzzo LV, Coombes KR. Pattern recognition in lymphoid malignancies using CytoGPS and Mercator. Under review.

Examples

```
x <- Chromosome(6)
image(x)
image(x, showBandNames = TRUE)
image(x, horiz = TRUE)
```

---

cytoband-data *Example Cytoband Data*

Description

This data set contains the genomic locations of cytobands along with loss-gain-fusion percentages for three group of samples.

Usage

```
data(cytoData)
```

Format

A data matrix (`cytobandLocations`) containing 868 rows and 14 columns. Each row contains one of the cytobands defined in the 2016 update to ISCN nomenclature. The rownames are the standard cytoband names; for example, `1p36.33`. The first five columns are the same as in the `cytobandLocations` data set. The remaining nine columns form three sets of three, recording the percentage of Loss, Gain, and Fusion events in three sets of samples, labeled "A", "B", and "C".

Author(s)

Kevin R. Coombes <krc@silicovore.com>, Dwayne G. Tally <dtally110@hotmail.com>

Source

The cytoband locations were downloaded from the UCSC Genome Browser and synchronized with the list of cytobands in ISCN 2016. The percentages were computed using tools in this package from subsets of the Mitelman Database of Chromosome Aberrations and Gene Fusions in Cancer.

References

J. McGowan-Jordan, A. Simons, M. Schmid (editors). ISCN 2016: An International System for Human Cytogenomic Nomenclature. Karger Publishing, Basel, 2016.

Mitelman, F., B. Johansson, and F. Mertens, Catalog of chromosome aberrations in cancer. Vol. 1. 1991: Wiley-Liss New York.

CytobandData-class *The* CytobandData *Class*

Description

A class to represent genome-wide data measured or summarized at cytoband-level resoluton.

Usage

```
CytobandData(data, info, genome = NULL)
## S4 method for signature 'CytobandData'
summary(object, ...)
```

Arguments

data	A data frame that must contain at least one numeric column and may contain the five cytoband location columns. If the latter are missing, they must be supplied by the genome argument.
info	A data frame with Label and Description columns. If missing, it will be created from the column names in the data argument.
genome	A data frame containing cytoband locations. Often, simply uses the cytobandLocations object describing build 38 of the human genome.
object	An object of the CytobandData class.
...	Additional arguments to the summary method; ignored.

Details

Conventional karyotyping describes chromosomal abnormalities (in a standardized text-based nomenclature) that are visible through a microsope. Karyotyping relies on a technique known as Giemsa staining, which creates a banding pattern along the chromosome of different shades of gray. The Loss-Gain-Fusion (LGF) model implemented at the http://cytogps.org web site converts text-based karyotypes into binary vectors, stored in JSON files. Our readLGF function parses the JSON files to create R data structures, including cytoband-resolution summaries of the frequencies of abnormalities. These summaries can be used to create CytobandData objects, which can be visualized using the barplot and image methods.

Value

The CytobandData conxtructor returns an object of the CytobandData class. The summary method returns a table, which is the usual return value when the summary method is applied to a data frame (in this case, the DATA slot).

Slots

DATA: A data frame containg five columns (Chromosome, loc.start, loc.end, Band, and Stain) defining chromosomal locations of cytobands, along with one or more columns of numerical data at cytoband resolution.

INFO: A data frame with two columns (Label and Description) that describes the columns of the DATA slot.

Methods

summary: signature(object = "RCytoGPS") Returns a summary of the DATA slot.

Author(s)

Kevin R. Coombes <krc@silicovore.com>, Dwayne G. Tally <dtally110@hotmail.com>

References

Abrams ZB, Tally DG, Coombes KR. RCytoGPS: An R Package for Analyzing and Visualizing Cytogenetic Data. In preparation.

Abrams ZB, Tally DG, Zhang L, Coombes CE, Payne PRO, Abruzzo LV, Coombes KR. Pattern recognition in lymphoid malignancies using CytoGPS and Mercator. Under review.

Examples

```
jsonDir <- system.file("Examples/JSONfiles", package = "RCytoGPS")
temp <- readLGF(folder = jsonDir)
cytoData <- data.frame(temp[["CL"]],temp[["frequency"]])
bandData <- CytobandData(cytoData)
class(bandData)
summary(bandData)
```

CytobandData-methods *Idiogram Graphs*

Description

The image and barplot methods of the CytobandData-class provide flexible displays of genome wide data that has been summarized at cytoband resolution.

Usage

```
## S4 method for signature 'CytobandData'
image(x, chr, what,
         pal = palette(), nrows = 2, labels = NULL,
         horiz = FALSE, axes = chr != "all", debug = FALSE, legend = FALSE,
         sigcolumn = NA, sigcut = 0.01, alpha = 63, clip = FALSE)
## S4 method for signature 'CytobandData'
barplot(height, what, col = "blue", altcol = "#FED4C4",
                  ylab = "Percent", hline = NULL,
                  xform = function(x) x, ...)
```

Arguments

x	An object of the CytobandData-class, which combines numerical data at cytoband resolution with information on the chromosomal locations of the cytobands.
chr	The specific chromosome you want to view, typically in c(1:22,"X","Y"). if you want to see all the chromosomes at once then you can set chr = "all".
what	A vector or list containing the names of the numerical column(s) that you want to display from the data frame. The barplot method only shows a single data column at a time. The plots resulting from the image method change depending on whether you supply a vector or a list, as well as on the length. For more details, see the vignettes, espefcialy the image gallery.
pal	a charcter vector containing the colors you want to use for different data shown in the plot.
horiz	A logical value determining whether images present the chromosome idiograms horizontally or vertically.
nrows	Only used when chr = "all" to determine the number of rows to use to display different chromosomes. Must be an integer between 1 and 4.
labels	Only use when what is a character vector (not a list) and chr is mot equal to "all". Used to label differnt columns of displayed data.
axes	Logical value; should axes be displayed?
legend	Logicla value; should a legend be added to the plot.

sigcolumn	The three parameters sigcolumn, sigcut, and alpha are used as a set. The first names a numerical column in the data set used to define "significance". The second is a vector of cutoffs that mark levels of significance. The latter is a number between 0 and 255 denoting the transparency level assigned to the color for each significance level.
sigcut	See sigcolumn.
alpha	See sigcolumn.
clip	A logical value; should te length of the chromosome fill the device (if TRUE) or be plotted relative to the length of Chromosome 1 (if FALSE). Currently only used when plotting two values, one on either side of a single chromosome.
debug	Logical value; should the method print out debugging information. Probably best to ignore.
height	An object of the CytobandData-class, which combines numerical data at cytoband resolution with information on the chromosomal locations of the cytobands.
col	a character vector containing the colors you want to use for different data shown in the plot.
altcol	Determines the color used, lternating with white, in the x-axis plot of all chromosomes.
ylab	Label for the y-axis in a barplot.
hline	Numeric vector of heights at which toadd a horizontal line.
xform	Function to transform the data before plotting. Defauly is the identity map, which does nothing.
...	Ignored.

Value

Both the image and barplot methods invisibly return their first argument, and object fo the CytobandData class.

Author(s)

Kevin R. Coombes <krc@silicovore.com>, Dwayne G. Tally <dtally110@hotmail.com>

References

Abrams ZB, Tally DG, Coombes KR. RCytoGPS: An R Package for Analyzing and Visualizing Cytogenetic Data. In preparation.

Abrams ZB, Tally DG, Zhang L, Coombes CE, Payne PRO, Abruzzo LV, Coombes KR. Pattern recognition in lymphoid malignancies using CytoGPS and Mercator. Under review.

Examples

```
jsonDir <- system.file("Examples/JSONfiles", package = "RCytoGPS")
x <- readLGF(folder = jsonDir)
cytoData <- data.frame(x[["CL"]], x[["frequency"]])
bandData <- CytobandData(cytoData)
datacolumns <- names(x[["frequency"]])
barplot(bandData, what = datacolumns[1], col="forestgreen")

image(bandData, what = datacolumns[1:3], chr = 2)
image(bandData, what = datacolumns[1:3], chr = "all")
image(bandData, what = as.list(datacolumns[1:2]), chr = 2)
image(bandData, what = as.list(datacolumns[1:2]), chr = "all")
```

---

cytobandLocations *Cytoband Locations*

Description

This data set contains the genomic locations of cytobands based on both the latest build of the human genome (GRch38) and the latest update to the International Standard for human Cytogenomic Nomenclature (ISCN). Note that the CRch38 locations are unchanged from build GRch37 (hg19).

Usage

```
data(cytobandLocations)
```

Format

A data matrix (`cytobandLocations`) containing 868 rows and 5 columns. Each row contains one of the cytobands defined in the 2016 update to ISCN nomenclature. The rownames are the standard cytoband names; for example, `1p36.33`. The columns are

Chromosome The name of the human chromosome, stored as `chr#`.

`loc.start` The starting base position of the band.

`loc.end` The ending base position of the band.

`Band` The band name without the chromosome; for example, `p36.33`.

`Stain` A factor containing the name of the Giemsa-stain color of the band in a karyotype image.

Also, a vector (`idiocolors`) of length eight that translates the Giemsa stain names into appropriate colors.

Author(s)

Kevin R. Coombes <krc@silicovore.com>, Dwayne G. Tally <dtally110@hotmail.com>

Source

The starting point for these data were the cytoband locations downloaded from the UCSC Genome Browser. We confirmed that the data were unchanged between human genome builds GRCh36 (hg18), GRCh37 (hg19), and GRCh38. However, the list of cytobands from UCSC was not consistent with the list of cytobands in ISCN 2016. We manually edited the source file to be compliant. It also matches the list of cytobands produced at http://cytogps.org.

References

J. McGowan-Jordan, A. Simons, M. Schmid (editors). ISCN 2016: An International System for Human Cytogenomic Nomenclature. Karger Publishing, Basel, 2016.

`preclean` *Pre-clean Karyotypes*

Description

A function to clean karyotype data, by deleting known comments that do not adhere to the ISCN standard.

Usage

```
preclean(x, targetColumns, dirt)
```

Arguments

`x`	A data frame containing at least one column of karyotype data.
`targetColumns`	Either a numeric vector of column indices or a character vector of column names.
`dirt`	A character vector containing items to delete from all karyotypes,

Details

The core input data worked on by the `RCytoGPS` are karyotypes, which are text strings written to conform to the ISCN standard. At many institutions, the cytogeneticists have developed idiosyncratic conventions that they use to add comments into the string. In most cases, these karyotypes are simply stored as text strings in a local database. In particular, they are not checked for synatx or grammar errors. By contrast, the implementation of the CytoGPS algorithm at the web site http://cytogps.org uses a formal approach with lexer and parser. As a result, many karyotypes are rejected by the system.

The `preclean` function uses `gsub` to delete a list of known (local) comments from all karyotypes, making it more likley that they will be successfully processed by the lexer and parser. We provide an example list derived from experience at our own institution.

Implementation Note: The `preclean` function removes strings in the order that they are contained in the `dirt` vector. So, you have to be carefully not to delete parts of a long phrase before trying to delete the whole phrase. For example, you should not remove "clonal" before removing "nonclonal".

Value

A data frame of te same size and with the same number of columns as the input data frame.

Author(s)

Kevin R. Coombes <krc@silicovore.com>

References

Abrams ZB, Zhang L, Abruzzo LV, Heerema NA, Li S, Dillon T, Rodriguez R, Coombes KR, Payne PRO. CytoGPS: a web-enabled karyotype analysis tool for cytogenetics. Bioinformatics. 2019 Dec 15;35(24):5365-5366.

Abrams ZB, Tally DG, Zhang L, Coombes CE, Payne PRO, Abruzzo LV, Coombes KR. Pattern recognition in lymphoid malignancies using CytoGPS and Mercator. In Press.

Abrams ZB, Tally DG, Coombes KR. RCytoGPS: An R Package for Analyzing and Visualizing Cytogenetic Data. In preparation.

Examples

```
cleanDir <- system.file("PreClean", package = "RCytoGPS")
bad <- read.delim(file.path(cleanDir, "badStrings.txt"), header=FALSE)
bad <- as.vector(as.matrix(bad))
input <- read.csv(file.path(cleanDir, "startKaryotypes.csv"))
myclean <- preclean(input, 4:5, bad)
```

---

readLGF *Extracting LGF karyotype data from JSON files*

Description

A function to read binary karyotype data, stored in LGF format in JSON files produced by the CytoGPS web site, into R for further analysis.

Usage

```
readLGF(files = NULL, folder = NULL, verbose = TRUE)
```

Arguments

files	The name of the JSON file (or a character vector of such file names) from which you want to extract and format data. If NULL, then it will extract all JSON files within the folder path provided.
folder	The specified directory/folder from which the user wants to extract JSON files. If NULL, then the function will look in the current working directory.
verbose	A logical value; should the function keep you informed about what it is doing?

Details

CytoGPS is an algorithm that converts conventional karyotypes from the standard text-based notation (the International Standard for Human Cytogenetic/Cytogenomic Nomenclature; ISCN) into binary vectors with three bits (loss, gain, or fusion) per cytoband, which we call the LGF model. The web site http://cytogps.org provides an implementation that allows users to upload text files containing one karyotype per line. It produces its output as a file in JavaScript Object Notation (JSON).

The `readLGF` function reads and parses these JSON files and converts them into an R data structure. The `raw` component of this structure contains binary matrices that can serve as input to the Mercator package (see `Mercator-class`) for unsupervised analyses. The `frequency` component summarizes the fraction of input karyotype-clones with each abnormality, and can be visualized with other tools in the `RCytoGPS` package.

Value

A list containing five elements:

- `source`: A character vector containing the names of the JSON files from which data was read.
- `raw`: A list of lists, one per JSON source file. Each internal list contains two elements, `Status` and `LGF`. `Status` is a data frame with one row per karyotype in the input file, describing the results of CytoGPS parsing and mapping. Results can be "`Success`", "`Nonfixable grammar error`", "`Validation error`", "`Fixable grammar error and success`", or "`Fixable grammar error but containing validation error`". `LGF` is a data frame where the columns are LGF-cytobands and the rows are clones from the successfully processed input karyotypes; each karyotype can have multiple clones. Entries are zero or one indicating the absence or presence of an abnormality.
- `frequency`: A data frame, where the rows are cytobands and the columns are Loss, Gain, and Fusion, with three columns per input file. Entries are the fraction of karyotype clones with that abnormality.
- `size`: An integer vector containing the total number of clones per input file. These values can be used to turn fractions back into counts.
- `CL`: A data frame with one row per cytoband detailing the chromosomal location and (grayscale) color of the band produced by Giemsa staining.

Author(s)

Kevin R. Coombes <krc@silicovore.com>, Dwayne G. Tally <dtally110@hotmail.com>

References

Abrams ZB, Zhang L, Abruzzo LV, Heerema NA, Li S, Dillon T, Rodriguez R, Coombes KR, Payne PRO. CytoGPS: a web-enabled karyotype analysis tool for cytogenetics. Bioinformatics. 2019 Dec 15;35(24):5365-5366.

Abrams ZB, Tally DG, Zhang L, Coombes CE, Payne PRO, Abruzzo LV, Coombes KR. Pattern recognition in lymphoid malignancies using CytoGPS and Mercator. Under review.

Abrams ZB, Tally DG, Coombes KR. RCytoGPS: An R Package for Analyzing and Visualizing Cytogenetic Data. In preparation.

See Also

Mercator-class

Examples

```
jsonDir <- system.file("Examples/JSONfiles", package = "RCytoGPS")
x <- readLGF(folder = jsonDir)

jsonFile <- dir(jsonDir, pattern = "*.json")[1]
y <- readLGF(jsonFile, jsonDir)
```

Index