

Package ‘contIBAIT’

October 12, 2016

Type Package

Title Improves Early Build Genome Assemblies using Strand-Seq Data

Version 1.0.0

Date 2016-02-02

Author Kieran O'Neill, Mark Hills, Mike Gottlieb

Maintainer Kieran O'Neill <koneill@bccrc.ca>

Description Using strand inheritance data from multiple single cells from the organism whose genome is to be assembled, contIBAIT can cluster unbridged contigs together into putative chromosomes, and order the contigs within those chromosomes.

License BSD_2_clause + file LICENSE

Depends BH (>= 1.51.0-3), Rsamtools (>= 1.21)

LinkingTo Rcpp, BH

Imports grDevices, clue, cluster, gplots, IRanges, GenomicRanges, S4Vectors, Rcpp, TSP, GenomicFiles, gtools, rtracklayer, BiocParallel, DNACopy, colorspace, reshape2, ggplot2, methods, exomeCopy, diagram

Suggests BiocStyle

biocViews CellBasedAssays, QualityControl, WholeGenome, Genetics, GenomeAssembly

Collate 'AllClasses.R' 'AllGenerics.R' 'BAIT.R'
'barplotLinkageGroupCalls.R' 'clusterContigs.R'
'computeConsensus.R' 'computeSim.R' 'contIBAIT.R'
'ideogramPlot.R' 'makeBoxPlot.R' 'makeChrTable.R'
'mapGapFromOverlap.R' 'mergeLinkageGroups.R'
'orderAllLinkageGroups.R' 'orderContigsGreedy.R'
'orderContigsTSP.R' 'plotContigOrder.R' 'plotLGDistances.R'
'plotWCdistribution.R' 'preprocessStrandTable.R'
'reorientLinkageGroups.R' 'strandSeqFreqTable.R' 'writeBed.R'

RoxygenNote 5.0.1

NeedsCompilation yes

R topics documented:

BAIT	3
barplotLinkageGroupCalls,LinkageGroupList,ChrTable-method	4
ChrTable-class	5
clusterContigs,StrandStateMatrix-method	5
contiBAIT	7
ContigOrdering-class	8
exampleChrTable	9
exampleCrickFreq	9
exampleDividedChr	9
exampleLGList	9
exampleReadCounts	9
exampleStrandFreq	10
exampleWatsonFreq	10
exampleWCMatrix	10
ideogramPlot,StrandReadMatrix,StrandReadMatrix,ChrTable-method	10
LinkageGroupList-class	11
makeBoxPlot,ChrTable,LinkageGroupList-method	12
makeChrTable	13
mapGapFromOverlap	14
mergeLinkageGroups,LinkageGroupList,StrandStateMatrix-method	14
orderAllLinkageGroups,LinkageGroupList,StrandStateMatrix,StrandFreqMatrix,StrandReadMatrix-method	15
orderContigsGreedy	16
orderContigsTSP	17
OrientationFrame-class	18
plotContigOrder	18
plotLGDistances,LinkageGroupList,StrandStateMatrix-method	19
plotWCdistribution,StrandFreqMatrix-method	20
preprocessStrandTable,StrandFreqMatrix-method	21
reorientLinkageGroups,LinkageGroupList,StrandStateMatrix-method	22
show,ContigOrdering-method	23
show,LinkageGroupList-method	23
show,OrientationFrame-method	24
show,StrandFreqMatrix-method	24
show,StrandReadMatrix-method	25
show,StrandStateMatrix-method	25
StrandFreqMatrix-class	26
StrandReadMatrix-class	26
strandSeqFreqTable	27
StrandStateMatrix-class	28
writeBed,ANY,OrientationFrame,ContigOrdering-method	29

BAIT	<i>BAIT – master function to process strand-seq libraries into BAIT ideograms</i>
------	---

Description

BAIT – master function to process strand-seq libraries into BAIT ideograms

Usage

```
BAIT(path = ".", splitBy = 2e+05, readQual = 10, pairedEnd = TRUE,
      plotBy = "lib", plotName = NULL, chroms = "all", verbose = TRUE)
```

Arguments

path	String denoting location of Strand-seq bam files (default is ".")
splitBy	numeric value of binsize for plotting. default is 200000
readQual	Integer dictating the minimal mapping quality required for a read to be accepted. Default is 10.
pairedEnd	Whether the bam files being read are in paired end format. Default is TRUE. Note,
plotBy	Whether to plot by library ('lib') or chromosome ('chr')
plotName	character which determines file name to be saved. Default is to open an R plot from the terminal
chroms	vector of chromosome number to prevent contig plotting. eg for humans use 1:24. Default is 'all'
verbose	prints messages to the terminal (default is TRUE)

Value

ideogram plots

Examples

```
## Not run:

#Not run because this takes a minute or two:
bamPath=file.path(system.file(package='contiBAIT'), 'extdata')

BAIT(bamPath, pairedEnd = FALSE) #our example data is single-end

## End(Not run)
```

`barplotLinkageGroupCalls,LinkageGroupList,ChrTable-method`

Bar plot all linkage groups, with the true chromosomes of contigs coloured.

Description

Bar plot all linkage groups, with the true chromosomes of contigs coloured.

Usage

```
## S4 method for signature 'LinkageGroupList,ChrTable'
barplotLinkageGroupCalls(object, chrTable,
  by = "lg", returnTable = FALSE)
```

Arguments

<code>object</code>	LinkageGroupList, as generated by <code>clusterContigs</code>
<code>chrTable</code>	GRanges object containing assembly information about the contigs, including a meta column called 'name' that has names matching the object. Note that the rownames of <code>chrTable</code> should be the contig names, as they are used in <code>object</code> , and the first column (chromosome name) will be used to order by chromosome if 'chr' option used in <code>by</code> parameter. To use a bam file header, the product of <code>makeChrTable(bamFile)</code> is suitable for input
<code>by</code>	whether to plot by linkage group (if 'lg') or chromosomes ('chr')
<code>returnTable</code>	TRUE to return chromosome length matrix. Note to include legend, use <code>legend=rownames(chr.table)</code> for <code>by='lg'</code> , and <code>legend=colnames(chr.table)</code> for <code>by='chr'</code>

Value

a matrix of lengths of each chromosome (rows) in each linkage group (columns)

Examples

```
data("exampleLGList")
data("exampleDividedChr")

barplotLinkageGroupCalls(exampleLGList, exampleDividedChr)
```

ChrTable-class *A class for storing chromosome/fragment lengths*

Description

This class is a GRanges object with a meta column called name, which represents the fragment name.

Constructor for ChrTable

Usage

```
ChrTable(chrRanges = GRanges())
```

Arguments

chrRanges a GRanges object with a meta column called name, which represents the fragment name

Value

a ChrTable

clusterContigs,StrandStateMatrix-method
clusterContigs – agglomeratively clusters contigs into linkage groups based on strand inheritance

Description

clusterContigs – agglomeratively clusters contigs into linkage groups based on strand inheritance

Usage

```
## S4 method for signature 'StrandStateMatrix'
clusterContigs(object, similarityCutoff = 0.7,
  recluster = NULL, minimumLibraryOverlap = 5, randomise = TRUE,
  randomSeed = NULL, randomWeight = NULL, clusterParam = NULL,
  clusterBy = "hetero", verbose = TRUE)
```

Arguments

object	data.frame containing strand inheritance information for every contig (rows) in every library (columns). This should be the product of strandSeqFreqTable
similarityCutoff	place contigs in a cluster when their strand state is at least this similar
recluster	Number of times to recluster and take the consensus of. If NULL, clustering is run only once.
minimumLibraryOverlap	for two contigs to be clustered together, the strand inheritance must be present for both contigs in at least this many libraries (in addition to their similarity being at least similarityCutoff)
randomise	whether to reorder contigs before clustering
randomSeed	random seed to initialize clustering
randomWeight	vector of weights for contigs for resampling. If NULL, uniform resampling is used. Typically this should be a measure of contig quality, such as library coverage, so that clustering tends to start from the better quality contigs.
clusterParam	optional BiocParallelParam specifying cluster to use for parallel execution. When NULL, execution will be serial.
clusterBy	Method for performing clustering. Default is 'hetero' (for comparing heterozygous calls to homozygous). Alternative is 'homo' (for comparison between the two homozygous calls)
verbose	prints function progress

Details

Note that a more stringent similarity cutoff will result in more clusters, and a longer run time, since at every iteration a distance is computed to the existing clusters. However, in lower-quality data, a more stringent cutoff may be necessary to reduce the number of contigs that are erroneously grouped.

Note that clusterParam

Value

LinkageGroupList of vectors containing labels of contigs belonging to each linkage group

Examples

```
data("exampleWCMatrix")

clusteredContigs <- clusterContigs(exampleWCMatrix, verbose=FALSE)
show(clusteredContigs)
show(clusteredContigs[[1]])

reorientedMatrix <- reorientLinkageGroups(clusteredContigs, exampleWCMatrix)
mergedLinkageGroups <- mergeLinkageGroups(clusteredContigs,reorientedMatrix[[1]])
```

contibAIT	<i>contibAIT – master function to process strand-seq libraries into ordered linkage groups</i>
-----------	--

Description

contibAIT – master function to process strand-seq libraries into ordered linkage groups

Usage

```
contibAIT(path = ".", cluster = 1, clusterParam = NULL,
          saveName = FALSE, filter = FALSE, readQual = 10, readLimit = 10,
          pairedEnd = TRUE, makePlots = FALSE, verbose = TRUE)
```

Arguments

path	String denoting location of Strand-seq bam files (default is ".")
cluster	Integer denoting the number of reclusterings to be performed for creating linkage groups (default is 1)
clusterParam	Number of parallel processors to use when clustering contigs. Default is NULL.
saveName	String denoting the file name for saved data. If FALSE, no intermediate files are saved (default is FALSE)
filter	additional file to split chromosomes based on locations. If this parameter is blank, a filter table will be automatically generated from the header of the first file in bamFileList
readQual	Integer dictating the minimal mapping quality required for a read to be accepted. Default is 10.
readLimit	Minimum number of reads on a contig to make a strand call. Default is 10
pairedEnd	Whether the bam files being read are in paired end format. Default is TRUE. Note, since paired reads will be the same direction, only first mate read of pair is used in output
makePlots	Logical determining whether plots should be created. Default is TRUE
verbose	prints messages to the terminal (default is TRUE)

Value

ordered contigs in bed format. Depending on options, intermediate files and plots will also be generated

Examples

```
#Get a list of BAM files containing libraries for cells from the same organism, aligned to the same genome
#In this case these are the example BAM files provided with the package (hence the call to system.file);
data("exampleDividedChr")
library(BiocParallel)

example.dir <- file.path(system.file(package='contigBAIT'), 'extdata')

orderedContigs <- contigBAIT(path=example.dir,
cluster=1,
clusterParam=MulticoreParam(),
filter=exampleDividedChr,
pairedEnd=FALSE)
```

ContigOrdering-class *A class for storing contig ordering of a linkage group*

Description

This class is a matrix of two character vectors that represent the calculated ordering of a linkage group. The first element of this matrix is the Linkage Group sub-setted by contigs with equal strand states across all libraries in the calculated order. The second element is the names of names of each contig in the calculated order.

Constructor for ContigOrdering

Usage

```
ContigOrdering(ordering = character())
```

Arguments

ordering a matrix of two character vectors that represent the calculated ordering of a linkage group. The first element of this matrix is the Linkage Group sub-setted by contigs with equal strand states across all libraries in the calculated order. The second element is the names of names of each contig in the calculated order.

Value

a ContigOrdering

Examples

```
thisOrdering <- ContigOrdering(matrix(ncol=2, c( "LG1.11", "chr2:1000820-2001640",
"LG1.1", "chr2:3002461-4003281")))
```

exampleChrTable	<i>Example of a ChromosomeTable, containing contigs and their lengths</i>
-----------------	---

Description

Example of a ChromosomeTable, containing contigs and their lengths

exampleCrickFreq	<i>An example Crick strand frequencies extracted from BAMS by strandSeqFreqTable where BAITables=TRUE</i>
------------------	---

Description

An example Crick strand frequencies extracted from BAMS by strandSeqFreqTable where BAITables=TRUE

exampleDividedChr	<i>Example of a divided chromosome, containing contigs and their lengths</i>
-------------------	--

Description

Example of a divided chromosome, containing contigs and their lengths

exampleLGList	<i>Example of a LinkageGroupList output from clusterContigs</i>
---------------	---

Description

Example of a LinkageGroupList output from clusterContigs

exampleReadCounts	<i>Example of read counts extracted from BAMS by strandSeqFreqTable</i>
-------------------	---

Description

Example of read counts extracted from BAMS by strandSeqFreqTable

exampleStrandFreq	<i>Example of strand frequencies extracted from BAMS by strandSeqFreqTable</i>
-------------------	--

Description

Example of strand frequencies extracted from BAMS by strandSeqFreqTable

exampleWatsonFreq	<i>An example Watson strand frequencies extracted from BAMS by strandSeqFreqTable where BAITables=TRUE</i>
-------------------	--

Description

An example Watson strand frequencies extracted from BAMS by strandSeqFreqTable where BAITables=TRUE

exampleWCMatrix	<i>An example StrandStateMatrix containing WW, CC and WC calls for contigs</i>
-----------------	--

Description

An example StrandStateMatrix containing WW, CC and WC calls for contigs

ideogramPlot,StrandReadMatrix,StrandReadMatrix,ChrTable-method	<i>ideogramPlot – plots BAIT-like ideograms</i>
--	---

Description

ideogramPlot – plots BAIT-like ideograms

Usage

```
## S4 method for signature 'StrandReadMatrix,StrandReadMatrix,ChrTable'
ideogramPlot(WatsonFreqList,
  CrickFreqList, chrTable, plotBy = "lib", showPage = FALSE,
  orderFrame = NULL, orientationData = NULL, verbose = TRUE)
```

Arguments

WatsonFreqList	data.frame of Watson calls. Product of strandSeqFreqTable[[3]] when BAITtables=TRUE
CrickFreqList	data.frame of Crick calls. Product of strandSeqFreqTable[[4]] when BAITtables=TRUE
chrTable	A data.frame consisting of chromosomes and lengths. Generated by makeChrTable(). Note rownames equal to chromosome names are required
plotBy	Whether to generate a plot for each library ('lib') or a plot for each chromosome ('chr')
showPage	Integer specifying which page to plot if plotBy='chr' selected. Useful when not plotting to a file. Default is FALSE
orderFrame	ordered data.frame of contigs (produced by orderAllLinkageGroups). Default is FALSE, where plots will be made from elements in chrTable.
orientationData	data.frame of contig orientations of type OrientationFrame telling which reads to flip Watson and Crick counts
verbose	prints messages to the terminal (default is TRUE)

Value

ordered contigs in bed format. Depending on options, intermediate files and plots will also be generated

Examples

```
data("exampleWatsonFreq")
data("exampleCrickFreq")
data('exampleDividedChr')

singleWatsonLibrary <- StrandReadMatrix(exampleWatsonFreq[,2, drop=FALSE])
singleCrickLibrary <- StrandReadMatrix(exampleCrickFreq[,2, drop=FALSE])

ideogramPlot(singleWatsonLibrary, singleCrickLibrary, exampleDividedChr)
```

LinkageGroupList-class

A class for storing linkage group calls for contigs

Description

This class is simply a list of character strings containing the names of linkage groups.

Constructor for LinkageGroupList

Usage

```
LinkageGroupList(linkageGroups = list(), names = character())
```

Arguments

linkageGroups a list of character vectors of names of contigs in each LG
 names a vector of names of linkage groups

Value

a LinkageGroupList

Examples

```
lgList <- LinkageGroupList(list(lg1=c('contig1', 'contig2'), lg2=c('contig3')),
names=c('lg1', 'lg20'))
```

makeBoxPlot,ChrTable,LinkageGroupList-method

makeBoxPlot – creates boxplot of contigs included in the analysis vs those excluded

Description

makeBoxPlot – creates boxplot of contigs included in the analysis vs those excluded

Usage

```
## S4 method for signature 'ChrTable,LinkageGroupList'
makeBoxPlot(chrTable, linkage.contigs)
```

Arguments

chrTable A GRanges object consisting of contigs and positions. A meta column called 'names' must be present with names in the same format as those from linkage.contigs. Generated by makeChrTable().

linkage.contigs A list of clustered contigs, generated by clusterContigs()

Value

a box plot of included and excluded contigs

Examples

```
#make an example barplot of data

data("exampleLGList")
data("exampleChrTable")
makeBoxPlot(exampleChrTable, exampleLGList)
```

makeChrTable	<i>makeChrTable – Pulls out chromosome and length data from the header of a bam file</i>
--------------	--

Description

makeChrTable – Pulls out chromosome and length data from the header of a bam file

Usage

```
makeChrTable(bamFile, splitFile = NULL, splitBy = NULL, verbose = TRUE)
```

Arguments

bamFile	string of location of a bam file to extract header data from
splitFile	GRanges object (of type chr, start and end: no strand or meta columns) of locations in which to split the assembly, such as previously determined locations of contig chimerism
splitBy	integer determining the average size contigs should be split by
verbose	if FALSE, no messages appear on terminal

Details

makeChrTable creates a table with chromosome name and chromosome length by extracting header data from the supplied bam file.

Value

a GRanges object of class ChrTable, containing information on the organism's chromosomes as extracted from the BAM file header.

Examples

```
#Get an example BAM file and generate a chromosome table featuring fragment names and lengths
example.bam <- list.files(file.path(system.file(package='contibAIT'), 'extdata'), full.names=TRUE)[1]

chrTable <- makeChrTable(example.bam)

show(chrTable)

dividedChr <- makeChrTable(example.bam, splitBy=1000000)

show(dividedChr)
```

mapGapFromOverlap	<i>mapGapFromOverlap</i> – function to co-localize strand state changes with assembly gaps
-------------------	--

Description

mapGapFromOverlap – function to co-localize strand state changes with assembly gaps

Usage

```
mapGapFromOverlap(sceFile, gapFile, chrTable, verbose = TRUE,
  overlapNum = 4)
```

Arguments

sceFile	GRanges object of strand state change locations in BED format
gapFile	GRanges object of assembly gaps in BED format (can be downloaded from UCSC table browser)
chrTable	GRanges object of chromosome table (product of makeChrTable)
verbose	prints messages to the terminal (default is TRUE)
overlapNum	Minimal number of strand state changes that overlap with a gap before assembly is cut at that location

Value

a GRanges object of all contigs split by regions where the sceFile and gapFile GRanges objects overlap.

mergeLinkageGroups,LinkageGroupList,StrandStateMatrix-method	<i>mergeLinkageGroups</i> – merge very similar linkage groups, including those in reverse orientation
--	---

Description

mergeLinkageGroups – merge very similar linkage groups, including those in reverse orientation

Usage

```
## S4 method for signature 'LinkageGroupList,StrandStateMatrix'
mergeLinkageGroups(object,
  allStrands, clusterParam = NULL, cluster = 1, similarityCutoff = 0.7)
```

*orderAllLinkageGroups,LinkageGroupList,StrandStateMatrix,StrandFreqMatrix,StrandReadMatrix-method*15

Arguments

object	LinkageGroupList
allStrands	StrandStateMatrix for all linkageGroups (usually reoriented by reorientStrandTable)
clusterParam	optional BiocParallelParam specifying cluster to use for parallel execution. When NULL, execution will be serial.
cluster	Integer denoting the number of reclusterings to be performed for creating linkage groups (default is 1)
similarityCutoff	merge contigs that are more similar this this

Value

list of indices within the allStrands matrix indicating linkage group membership, a list consisting of a strandStateMatrix (a reoriented version of allStrands), and a data.frame of type OrientationFrame containing contig names and orientations, as '+' or '-'.

Examples

```
data(exampleWCMatrix)
clusteredContigs <- clusterContigs(exampleWCMatrix, randomise=FALSE)

reorientedMatrix <- reorientLinkageGroups(clusteredContigs,
  exampleWCMatrix)

exampleLGLList <- mergeLinkageGroups(clusteredContigs,
  reorientedMatrix[[1]])
```

orderAllLinkageGroups,LinkageGroupList,StrandStateMatrix,StrandFreqMatrix,StrandReadMatrix-method
Function to call contig ordering algorithms iteratively across each linkage group element

Description

Function to call contig ordering algorithms iteratively across each linkage group element

Usage

```
## S4 method for signature
## 'LinkageGroupList,StrandStateMatrix,StrandFreqMatrix,StrandReadMatrix'
orderAllLinkageGroups(linkageGroupList,
  strandStateMatrix, strandFreqMatrix, strandReadCount, whichLG = NULL,
  saveOrdered = FALSE, orderCall = "greedy", randomAttempts = 75,
  verbose = TRUE)
```

Arguments

linkageGroupList	list of vectors, each specifying which contigs belong in which linkage group (product of clusterContigs)
strandStateMatrix	table of strand calls for all contigs (product of preprocessStrandTable)
strandFreqMatrix	table of W:C read proportions (used for QC) (product of strandSeqFreqTable[[1]])
strandReadCount	table of read counts (product of strandSeqFreqTable[[2]])
whichLG	vector of integers specifying the element(s) of linkageGroupList to be ordered (i.e. which specific linkage groups to try to order). Default is all LGs.
saveOrdered	Will return a pdf of heatmaps for each linkage group; String entered becomes the fileName (default is saveOrderedPDF=FALSE)
orderCall	currently either 'greedy' for greedy algorithm or 'TSP' for travelling salesperson algorithm (default is 'greedy')
randomAttempts	integer specifying number of randomized clusterings to identify the best ordering. Default is 75
verbose	Prints messages to the terminal. Default is TRUE

Value

a data.frame of ordered contigs with linkage group names

Examples

```
#Get a data.frame of ordered contigs from cells from the same organism, aligned to the same genome

data("exampleLGList")
data("exampleWCMatrix")
data("exampleStrandFreq")
data("exampleReadCounts")
contigOrder <- orderAllLinkageGroups(exampleLGList, exampleWCMatrix, exampleStrandFreq, exampleReadCounts)

show(contigOrder)
```

orderContigsGreedy	<i>Function to order contigs within a single linkage group using a greedy algorithms Attempt to order contigs within</i>
--------------------	--

Description

Function to order contigs within a single linkage group using a greedy algorithms Attempt to order contigs within

Usage

```
orderContigsGreedy(linkageGroupReadTable, randomAttempts = 75,
  verbose = TRUE)
```

Arguments

linkageGroupReadTable
dataframe of strand calls (product of combineZeroDists or preprocessStrandTable)

randomAttempts number of times to repeat the greedy algorithm with a random restart

verbose whether to print verbose messages

Value

list of two members: 1) contig names in order, 2) the original data.frame entered into function correctly ordered

orderContigsTSP	<i>Attempt to order contigs within linkage groups using travelling salesperson algorithm</i>
-----------------	--

Description

Attempt to order contigs within linkage groups using travelling salesperson algorithm

Usage

```
orderContigsTSP(linkageGroupReadTable)
```

Arguments

linkageGroupReadTable
dataframe of strand calls (product of combineZeroDists or preprocessStrandTable)

Value

list of two members: 1) contig names in order, 2) the original data.frame entered into function correctly ordered

OrientationFrame-class

A class for storing contig orientations

Description

This class is a matrix of two character vectors that represent the orientation of contigs. The first element of thismatrix is the contigs name The second element is the orinetation (as either + or -).

Constructor for OrientationFrame

Usage

```
OrientationFrame(orientation = character())
```

Arguments

`orientation` a matrix of two character vectors that represent the orientation of contigs. The first element of thismatrix is the contigs name The second element is the orinetation (as either + or -).

Value

a OrientationFrame

Examples

```
OrientationFrame(matrix(ncol=2, c("chr4:3002423-4003230", "+",
"chr4:140113083-141113889", "+")))
```

plotContigOrder

Plot ordering of contigs within a single linkage group.

Description

Plot ordering of contigs within a single linkage group.

Usage

```
plotContigOrder(contigOrder, lg)
```

Arguments

`contigOrder` matrix from orderAllContigs with the subdivided linkage groups and the names of the contigs to plot

`lg` Integer specifying the linkage group by which to plot

Value

A ggplot object (which will be plotted automatically if not assigned).

Examples

#Get a data.frame of ordered contigs from cells from the same organism, aligned to the same genome

```
data("exampleLGList")
data("exampleWCMatrix")
data("exampleStrandFreq")
data("exampleReadCounts")
contigOrder <- orderAllLinkageGroups(exampleLGList,
  exampleWCMatrix,
  exampleStrandFreq,
  exampleReadCounts)
plotContigOrder(contigOrder)
```

plotLGDistances,LinkageGroupList,StrandStateMatrix-method

plotLGDistances – plots a heatmap of the distances between linkage groups

Description

plotLGDistances – plots a heatmap of the distances between linkage groups

Usage

```
## S4 method for signature 'LinkageGroupList,StrandStateMatrix'
plotLGDistances(object,
  allStrands, lg = "all", labels = TRUE)
```

Arguments

object	LinkageGroupList
allStrands	StrandStateMatrix for all linkageGroups (usually reoriented by reorientStrandTable)
lg	= 'all' vector of integers to determine which linkage group(s) to plot. 'all' will calculate consensus strand calls for all linkage groups and plot them side by side (default it 'all')
labels	= TRUE if TRUE, contig names will be plotted on the axes
...	additional parameters to pass to heatmap.2

Value

a heatmap of linkage group calls

Examples

```
data("exampleLGList")
data("exampleWCMatrix")

plotLGDistances(exampleLGList, exampleWCMatrix)
plotLGDistances(exampleLGList, exampleWCMatrix, lg=1)
```

```
plotWCdistribution,StrandFreqMatrix-method
```

Creates median distribution boxplots across all libraries and contigs

Description

Creates median distribution boxplots across all libraries and contigs

Usage

```
## S4 method for signature 'StrandFreqMatrix'
plotWCdistribution(object, filterThreshold = 0.8)
```

Arguments

object	object of class StrandFreqMatrix (product of strandSeqFreqTable)
filterThreshold	numeric value used in assessing the threshold for homozygous strand calls. Default is 0.8.

Value

nothing, just plots.

Examples

```
data("exampleStrandFreq")

plotWCdistribution(exampleStrandFreq, filterThreshold=0.8)
```

```
preprocessStrandTable,StrandFreqMatrix-method
    preprocessStrandTable – remove low quality libraries and contigs be-
    fore attempting to build a genome
```

Description

preprocessStrandTable – remove low quality libraries and contigs before attempting to build a genome

Usage

```
## S4 method for signature 'StrandFreqMatrix'
preprocessStrandTable(strandTable,
  strandTableThreshold = 0.8, filterThreshold = 0.8,
  orderMethod = "libsAndConc", lowQualThreshold = 0.9, verbose = TRUE,
  minLib = 10, ignoreInternalQual = FALSE)
```

Arguments

strandTable	data.frame containing the strand table to use as input
strandTableThreshold	threshold at which to call a contig WW or CC rather than WC
filterThreshold	maximum number of libraries a contig can be NA or WC in
orderMethod	the method to order contigs. currently libsAndConc only option. Set to FALSE to not order contigs based on library quality
lowQualThreshold	background threshold at which to toss an entire library
verbose	messages written to terminal
minLib	minimum number of libraries a contig must be present in to be included in the output
ignoreInternalQual	logical that prevents function for making an overall assessment of library quality. Very chimeric assemblies can appear low quality across all libraries.

Value

A list of one matrix and three quality data.frames – 1: a matrix of WW/WC/WW calls for all contigs; 3: the quality of libraries used (based on frequencies outside expected ranges); 4: A data.frame of libraries that are of low quality and therefore excluded from analysis; 5: contigs that are present as WC in more libraries than expected. These are excluded from the strandStateMatrix, but are potentially worth investigating for chimerism.

Examples

```
data("exampleStrandFreq")

strandStates <- preprocessStrandTable(exampleStrandFreq, lowQualThreshold=0.8)

show(strandStates[[1]]) # WW-WC-CC matrix
```

```
reorientLinkageGroups,LinkageGroupList,StrandStateMatrix-method
reorientLinkageGroups uses a simple dissimilarity to find misoriented
fragments within linkage groups.
```

Description

`reorientLinkageGroups` uses a simple dissimilarity to find misoriented fragments within linkage groups.

Usage

```
## S4 method for signature 'LinkageGroupList,StrandStateMatrix'
reorientLinkageGroups(object,
  allStrands, previousOrient = NULL, verbose = TRUE)
```

Arguments

<code>object</code>	List of vectors containing names of contigs belonging to each LG.
<code>allStrands</code>	Table of type <code>strandStateMatrix</code> encompassing strand state for all contigs. Product of <code>StrandSeqFreqTable</code> .
<code>previousOrient</code>	data.frame of type <code>OrientationFrame</code> of previous orientation states if performed. Default is <code>NULL</code>
<code>verbose</code>	Outputs information to the terminal. Default is <code>TRUE</code> .

Value

a list consisting of a `strandStateMatrix` (a reoriented version of `allStrands`), and a data.frame of type `OrientationFrame` containing contig names and orientations, as '+' or '-'.

Examples

```
data(exampleWCMatrix)
clusteredContigs <- clusterContigs(exampleWCMatrix, randomise=FALSE)

reorientedMatrix <- reorientLinkageGroups(clusteredContigs,
  exampleWCMatrix)

# Note that in this example data, everything is correctly oriented to
# to begin with, so all contigs come out as + orientation
```

show,ContigOrdering-method
show-methods

Description

Shows a ContigOrdering

Usage

```
## S4 method for signature 'ContigOrdering'  
show(object)
```

Arguments

object a ContigOrdering

Value

nothing

show,LinkageGroupList-method
show-methods

Description

Shows a LinkageGroupList

Usage

```
## S4 method for signature 'LinkageGroupList'  
show(object)
```

Arguments

object a LinkageGroupList

Value

nothing

show,OrientationFrame-method
show-methods

Description

Shows a OrientationFrame

Usage

```
## S4 method for signature 'OrientationFrame'  
show(object)
```

Arguments

object a OrientationFrame

Value

nothing

show,StrandFreqMatrix-method
show-methods

Description

Shows a StrandFreqMatrix

Usage

```
## S4 method for signature 'StrandFreqMatrix'  
show(object)
```

Arguments

object a StrandFreqMatrix

Value

nothing

show,StrandReadMatrix-method
show-methods

Description

Shows a StrandReadMatrix

Usage

```
## S4 method for signature 'StrandReadMatrix'  
show(object)
```

Arguments

object a StrandReadMatrix

Value

nothing

show,StrandStateMatrix-method
show-methods

Description

Shows a StrandStateMatrix

Usage

```
## S4 method for signature 'StrandStateMatrix'  
show(object)
```

Arguments

object a StrandStateMatrix

Value

nothing

 StrandFreqMatrix-class

A class for storing a matrix of frequencies of Watson to Crick reads for a set of contigs over several libraries

Description

The strand information stored in this object is the ratio of Watson to Crick reads mapping to each contig in each library (cell). This should fall within the range (-1,1). This class simply extends matrix, but with additional validity checking.

Constructor for StrandFreqMatrix

Usage

```
StrandFreqMatrix(counts = matrix(double()))
```

Arguments

counts a double matrix of read count ratios

Value

a StrandFreqMatrix

Examples

```
data("exampleWatsonFreq")
data("exampleCrickFreq")
frequencyMatrix <- sapply(1:ncol(exampleCrickFreq),
  function(colNum){exampleCrickFreq[,colNum] / exampleWatsonFreq[,colNum]})

StrandFreqMatrix(frequencyMatrix)
```

StrandReadMatrix-class

A class for storing read counts for a set of contigs over several libraries

Description

The information stored in this class is simple read counts, so should be integers ≥ 0 .

Constructor for StrandReadMatrix

Usage

```
StrandReadMatrix(counts = matrix(integer()))
```

Arguments

counts an integer matrix of read counts

Value

a StrandReadMatrix

Examples

```
data("exampleWatsonFreq")
StrandReadMatrix(exampleWatsonFreq[,2, drop=FALSE])
```

strandSeqFreqTable *strandSeqFreqTable – function to process bam files for contiBAIT*

Description

strandSeqFreqTable – function to process bam files for contiBAIT

Usage

```
strandSeqFreqTable(bamFileList, fieldSep = ".", field = 1, qual = 0,
  rmdup = TRUE, verbose = TRUE, filter = NULL, tileChunk = 1e+05,
  pairedEnd = TRUE, BAITtables = FALSE)
```

Arguments

bamFileList	vector containing the location of the bams file to be read
fieldSep	The field separator of the bam file to use to define the field. Default is '.'
field	The field of the bam file name to use as an index (default is 1)
qual	Mapping quality threshold. Default is 0
rmdup	remove duplicates in output file. Default is TRUE
verbose	prints messages to the terminal (default is TRUE)
filter	additional file of type GRanges (with a meta column titled 'name' determining contig name) to split chromosomes based on locations. If this parameter is blank, a filter table will be automatically generated from the header of the first file in bamFileList.
tileChunk	Number of reads to split bam files into (smaller number requires less RAM). Default is 100000.
pairedEnd	Whether the bam files being read are in paired end format. Default is TRUE. Note, since paired reads will be the same direction, only first mate read of pair is used in output
BAITtables	creates additional matrices in the returned list with just Watson and Crick read counts to be used in downstream BAIT plotting. Default is FALSE

Value

a list containing two matrices: a StrandFreqMatrix of W:C read frequencies, and a StrandReadMatrix of read counts

Examples

```
#Get a list of BAM files containing libraries for cells from the same organism, aligned to the same genome
#In this case these are the example BAM files provided with the package (hence the call to system.file);
```

```
example.dir <- file.path(system.file(package='contiBAIT'), 'extdata')
bam.files <- dir(example.dir, full.names=TRUE)
```

```
strand.freq <- strandSeqFreqTable(bam.files, pairedEnd = FALSE)
```

```
show(strand.freq[[1]])
show(strand.freq[[2]])
```

StrandStateMatrix-class

A class for storing a data frame of discrete strand states of a set of contigs over several libraries

Description

The strand information stored in this object is a call of the strand state of each contig in each library. mapping to each contig in each library (cell). This should fall within the range (-1,1). This class simply extends matrix, but with additional validity checking.

Constructor for StrandStateMatrix

Usage

```
StrandStateMatrix(states = matrix(integer()))
```

Arguments

states an integer matrix of strand states by library

Value

a StrandStateMatrix

Examples

```
StrandStateMatrix(matrix(ncol=2, c(1,3,1,2)))
```

```
writeBed,ANY,OrientationFrame,ContigOrdering-method
      function to write contig order to BED file
```

Description

function to write contig order to BED file

Usage

```
## S4 method for signature 'ANY,OrientationFrame,ContigOrdering'
writeBed(chrTable,
         orientationData, contigOrder, libWeight = NULL,
         file = "contiBAIT_assembly.bed")
```

Arguments

chrTable	a GRanges object with a 'name' meta column matching contig names. Product of makeChrTable
orientationData	data.frame of contig and strand (with rownames matching contig names). Product of reorientLinkageGroups[[2]]
contigOrder	an object of type ContigOrdering with ordered Linkage Groups and contigs. Product of orderAllLinkageGroups
libWeight	average quality across all libraries for a contig
file	character string for bed file name to write

Value

NULL; BED file written to file

Examples

```
## Not run:

data("exampleDividedChr")

writeBed(exampleDividedChr,
         reorientedMatrix[[2]],
         contigOrder)

## End(Not run)
```

Index

*Topic **data**

- exampleChrTable, 9
 - exampleCrickFreq, 9
 - exampleDividedChr, 9
 - exampleLGList, 9
 - exampleReadCounts, 9
 - exampleStrandFreq, 10
 - exampleWatsonFreq, 10
 - exampleWCMatrix, 10
- BAIT, 3
- barplotLinkageGroupCalls
 - (barplotLinkageGroupCalls, LinkageGroupList-method), 4
- barplotLinkageGroupCalls, LinkageGroupList, ChrTable-method, 4
- barplotLinkageGroupCalls, LinkageGroupList, LinkageGroupList-method, ChrTable, ChrTable-method
 - (barplotLinkageGroupCalls, LinkageGroupList, ChrTable-method), 4
- ChrTable (ChrTable-class), 5
- ChrTable-class, 5
- clusterContigs
 - (clusterContigs, StrandStateMatrix-method), 5
- clusterContigs, StrandStateMatrix, StrandStateMatrix-method, 5
- clusterContigs, StrandStateMatrix-method, 5
- contiBAIT, 7
- ContigOrdering (ContigOrdering-class), 8
- ContigOrdering-class, 8
- exampleChrTable, 9
- exampleCrickFreq, 9
- exampleDividedChr, 9
- exampleLGList, 9
- exampleReadCounts, 9
- exampleStrandFreq, 10
- exampleWatsonFreq, 10
- exampleWCMatrix, 10
- ideogramPlot
 - (ideogramPlot, StrandReadMatrix, StrandReadMatrix, ChrTable-method), 10
- ideogramPlot, StrandReadMatrix, StrandReadMatrix, ChrTable-method, 10
- ideogramPlot, StrandReadMatrix, StrandReadMatrix-method, ChrTable-method, 10
- ideogramPlot, StrandReadMatrix, StrandReadMatrix, ChrTable-method, 10
- makeBoxPlot, ChrTable, LinkageGroupList-method, 12
- makeBoxPlot-ChrTable-LinkageGroupList-method
 - (makeBoxPlot, ChrTable, LinkageGroupList-method), 12
- makeChrTable, 13
- mergeLinkageGroups, 14
- mergeLinkageGroups, LinkageGroupList, StrandStateMatrix-method, 14
- mergeLinkageGroups-LinkageGroupList-StrandStateMatrix-method, 14
- mergeLinkageGroups, LinkageGroupList, StrandStateMatrix-method, 14
- orderAllLinkageGroups
 - (orderAllLinkageGroups, LinkageGroupList, StrandStateMatrix-method), 15
- orderAllLinkageGroups, LinkageGroupList, StrandStateMatrix-method, 15

orderAllLinkageGroups, orderAllLinkageGroups-LinkageGroupList-StrandStateMatrix-StrandFreqMatrix-StrandReadMatrix-method, 25
 (orderAllLinkageGroups, LinkageGroupList, StrandStateMatrix, StrandFreqMatrix, StrandReadMatrix-method), 15
 (StrandFreqMatrix-class), 26
 orderContigsGreedy, 16 StrandFreqMatrix-class, 26
 orderContigsTSP, 17 StrandReadMatrix
 OrientationFrame (StrandReadMatrix-class), 26
 (OrientationFrame-class), 18 StrandReadMatrix-class, 26
 OrientationFrame-class, 18 strandSeqFreqTable, 27
 StrandStateMatrix
 (StrandStateMatrix-class), 28
 plotContigOrder, 18 StrandStateMatrix-class, 28
 plotLGDistances (plotLGDistances, LinkageGroupList, StrandStateMatrix-method), 19
 writeBed
 plotLGDistances, LinkageGroupList, LinkageGroupList-method, writeBed, ANY, OrientationFrame, ContigOrdering-method, 19
 (plotLGDistances, LinkageGroupList, StrandStateMatrix-method), 19
 writeBed, ANY, OrientationFrame, ContigOrdering-method, 19
 plotLGDistances, LinkageGroupList, StrandStateMatrix-method, writeBed-OrientationFrame-ContigOrdering-method, 19
 (writeBed, ANY, OrientationFrame, ContigOrdering-method), 29
 plotWCdistribution (plotWCdistribution, StrandFreqMatrix-method), 20
 20
 plotWCdistribution, StrandFreqMatrix, StrandFreqMatrix-method (plotWCdistribution, StrandFreqMatrix-method), 20
 20
 plotWCdistribution, StrandFreqMatrix-method, 20
 20
 preprocessStrandTable (preprocessStrandTable, StrandFreqMatrix-method), 21
 21
 preprocessStrandTable, StrandFreqMatrix, StrandFreqMatrix-method (preprocessStrandTable, StrandFreqMatrix-method), 21
 21
 preprocessStrandTable, StrandFreqMatrix-method, 21
 21
 reorientLinkageGroups (reorientLinkageGroups, LinkageGroupList, StrandStateMatrix-method), 22
 22
 reorientLinkageGroups, LinkageGroupList, StrandStateMatrix-method, 22
 22
 reorientLinkageGroups-LinkageGroupList-StrandStateMatrix-method (reorientLinkageGroups, LinkageGroupList, StrandStateMatrix-method), 22
 22
 show, ContigOrdering-method, 23
 show, LinkageGroupList-method, 23
 show, OrientationFrame-method, 24
 show, StrandFreqMatrix-method, 24
 show, StrandReadMatrix-method, 25