Package 'Repliscope'

October 12, 2022

Type Package

Language en-GB

Title Replication Timing Profiling using DNA Copy Number

Version 1.1.1

Author Dzmitry G Batrakou

Maintainer Dzmitry G Batrakou <d.batrakou@gmail.com>

Description Create, Plot and Compare Replication Timing Profiles. The method is described in Muller et al., (2014) <doi:10.1093/nar/gkt878>.

License GPL-3

Encoding UTF-8

LazyData true

RoxygenNote 7.2.1

Depends R (>= 3.5.0), ggplot2, shiny

Imports stats, utils, methods, grDevices, colourpicker

Suggests knitr, rmarkdown

VignetteBuilder knitr

NeedsCompilation no

Repository CRAN

Date/Publication 2022-09-13 07:20:02 UTC

R topics documented:

`calcTrep` *A function to calculate Trep values from a sync-seq experiment calcTrep function fits a Boltzman sigmoid function into relative copy number datapoints for every genomic bin of the provided sync-seq merged dataframe. It then extracts time at which half of the cells have this genomic bin replicated (Trep). The output of the function is a dataframe containing Trep and TrepErr data for every genomic bin in a BED-like format.*

Description

A function to calculate Trep values from a sync-seq experiment calcTrep function fits a Boltzman sigmoid function into relative copy number datapoints for every genomic bin of the provided sync-seq merged dataframe. It then extracts time at which half of the cells have this genomic bin replicated (Trep). The output of the function is a dataframe containing Trep and TrepErr data for every genomic bin in a BED-like format.

Usage

```
calcTrep(ratioDFs, times)
```

Arguments

`ratioDFs`	A merged ratios dataframe containing sync-seq samples (dataframe).
`times`	Time series data in the same order as in the ratioDFs (numeric vector).

Examples

```
TrepDF <- calcTrep(subset(syncSeq[["data"]],chrom=="chrI"),times=c(25,30,35,40,45,50,90))
```

`compareRatios` *A function to compare two replication profiles*

Description

compareRatios takes two ratio dataframes that were binned the same way and uses z-score statistics to find p-values of their differences. The function outputs a combined dataframe containing the two input ratio dataframes in a long format with added 'p.value' column.

Usage

```
compareRatios(ratio1, ratio2)
```

Arguments

`ratio1`	Ratio dataframe, or a string containing name of a ratio dataframe (dataframe or string).
`ratio2`	Ratio dataframe, or a string containing name of a ratio dataframe (dataframe or string).

Examples

```
ratioDFs <- compareRatios(W303norm,Dbf4myc)
```

`Dbf4myc` *Sequence read coverage ratios for S.cerevisiae Dbf4-9myc sample.*

Description

Sequence read coverage ratios for Dbf4-9myc sample (T9394 strain). The cells were stained with DNA dye and sorted based on DNA content into S or G2/M phase fractions. Extracted DNA was sequenced and mapped to sacCer3 genome. Unique reads for replicating (S) and non-replicating (G2/M) samples were calculated in 1 kb genomic bins. The ratio was created by dividing 'score' values from replicating sample by non- replicating sample 'score' values, adjusted by total number of reads. The ratio values were further adjusted by multiplying them by 1.402 to put the values onto biologically relevant relative copy number scale from 1 to 2.

Usage

```
data(Dbf4myc)
```

Format

data frame with 11356 rows and 7 variables:

chrom short chromosome name
chromStart left chromosome coordinate
chromEnd right chromosome coordinate
name.rep replicating sample name
name.nonRep non-replicating sample name
ratio ratio value in the current bin
ratioFactor adjustment factor used for the current ratio

Source

S phase sample: SRA; G2 sample: SRA

References

Natsume et al. (2013) Mol Cell 50(5):661-74 (PubMed)

Examples

```
data(Dbf4myc)
```

`guide` *Guide dataframe for plotting smoothed sortSeq data*

Description

Guide dataframe for plotting smoothed sortSeq data

Usage

```
data(guide)
```

Format

Dataframe with 2 rows and 6 variables:

order Order to plot data in
name.rep Name of replicating sample
name.nonRep Name of non-replicating sample
raw Should raw data be plotted?
smooth Should smooth data be plotted?
color Color to plot the profile in

Examples

```
data(guide)
```

loadBed *Load a BED formatted file.*

Description

The function reads BED formatted files. The BED file format defined by UCSC: http://genome.ucsc.edu/FAQ/FAQformat. First three columns ("chrom", "chromStart", "chromEnd") are mandatory. The file fields may be separated by tabs, spaces or commas. If the BED file contains a header, it will be ignored. If a genome mask dataframe is provided, only data intersected with the mask will be retained. Resulting data is ordered by "chromStart" columns.

Usage

```
loadBed(file, genome = NULL, name = NULL)
```

Arguments

file	Path to the BED file (string, mandatory)
genome	A mask dataframe to exclude data from the BED file (dataframe, optional). The genome dataframe must contain "chrom" column and may further contain "chromStart" and "chromEnd" columns in this order.
name	A string to replace the 'name' column of the loaded BED file with (string, optional).

Examples

```
W303_G2 <- loadBed(system.file("extdata/W303_G2.bed",package="Repliscope"), name='W303_G2')
W303_G2_chrI <- loadBed(system.file("extdata/W303_G2.bed",package="Repliscope"),
                        name='W303_G2',genome=sacCer3[["genome"]])
```

makeGenome *A helper function to create a gemome dataframe*

Description

makeGenome is called by plotGenome() and plotCoverage() functions if a genome dataframe is not provided. It creates a BED-like dataframe containing unique chromosome names, their start coordinates (assumed 0), their ends (highest value in the corresponding 'chromEnd' of the BED dataframe) and 'midY' column containing half the max value of the "score" or "ratio" columns per chromosome. This later serves as y coordinate to add chromosome names. Axis name is passed via comment to the output dataframe. Extract it with 'attributes(genome)$axisName'.

Usage

```
makeGenome(DF, region = FALSE)
```

Arguments

DF	A BED or ratio dataframe containing either 'score' or 'ratio' column (dataframe).
region	String in the format 'chrI:1000-3000' (string, optional).

Examples

```
genomeDF <- makeGenome(W303_G2)
```

---

makeLabels *A helper function to create axis ticks and human readable labels.*

Description

makeLabels is called by plotGenome() and plotCoverage() functions. It creates a dataframe containing two columns: 'ticks' and 'labels'. 'Ticks' contains axis ticks coordinates, 'labels' will contain human readable lables for the ticks (using prefixes and optional units).

Usage

```
makeLabels(theMin, theMax, unit = "")
```

Arguments

theMin	Minimum value for the scale (double).
theMax	Maximum value for the scale (double).
unit	Unit to use for the labels (string, optional)

Examples

```
labels <- makeLabels(0,1200000,"b")
```

---

makeRatio *A function to calculate 'score' ratio between two bed dataframes makeRatio merges two supplied bed dataframes, calculates ratio of their "score" values normalises the ratio by the 'score' sums.*

Description

A function to calculate 'score' ratio between two bed dataframes makeRatio merges two supplied bed dataframes, calculates ratio of their "score" values normalises the ratio by the 'score' sums.

Usage

```
makeRatio(bedRep, bedNonRep)
```

Arguments

bedRep	Bed dataframe containing read counts from a replicating sample (dataframe). The bed dataframe must contain "chrom","chromStart", "chromEnd" and "score" columns.
bedNonRep	Bed dataframe containing read counts from a non-replicating sample (dataframe). The bed dataframe must contain "chrom","chromStart", "chromEnd" and "score" columns.

Examples

```
ratioDF <- makeRatio(W303_S,W303_G2)
```

MFAseq — *Replication profile for wild type DS2 H.volcanii*

Description

Replication profile for H.volcanii wild isolate DS2. Genomic DNA for deep sequencing was isolated from 100 ml culture in stationary phase (A650 > 1, DS2_stat sample) or 1 litre in exponential phase (A650 0.1, DS2_exp sample). Unique reads for the two samples were calculated in 1 kb genomic bins using ASM2568v1 genome assembly. The ratio was created by dividing 'score' values from replicating sample by non- replicating sample 'score' values, adjusted by total number of reads. The ratio values were further adjusted by multiplying them by 1.12 to put the values onto biologically relevant relative copy number scale from 1 to 2.

Usage

```
data(MFAseq)
```

Format

data frame with 3887 rows and 7 variables:

chrom short chromosome name

chromStart left chromosome coordinate

chromEnd right chromosome coordinate

name.rep replicating sample name

name.nonRep non-replicating sample name

ratio ratio value in the current bin

ratioFactor adjustment factor used for the current ratio

Source

DS2_exp exponential phase sample: SRA; DS2_stat stationary sample: SRA

References

Hawkins et al. (2013) Nature 503(7477):544-547 (PubMed)

Examples

```
data(MFAseq)
```

`normaliseRatio` *A function to normalise ratio values from 'ratio' column of the provided dataframe to fit biologically-relevant scale. It scales values either using supplied 'rFactor' value or automatically to best fit 1 to 2 scale (the upper limit of the scale may be adjusted with the upperLimit parameter). Normalisation factor used is stored in 'ratioFactor' column and also passed as the dataframe comment. To extract it, use 'attributes(mergedBed)$comment'*

Description

A function to normalise ratio values from 'ratio' column of the provided dataframe to fit biologically-relevant scale. It scales values either using supplied 'rFactor' value or automatically to best fit 1 to 2 scale (the upper limit of the scale may be adjusted with the upperLimit parameter). Normalisation factor used is stored in 'ratioFactor' column and also passed as the dataframe comment. To extract it, use 'attributes(mergedBed)$comment'

Usage

```
normaliseRatio(ratioDF, rFactor = NULL, upperLimit = 2, replace = TRUE)
```

Arguments

`ratioDF`	A ratio dataframe containing 'ratio' column (dataframe).
`rFactor`	Value to normalise by, related to replication progression (numeric, optional).
`upperLimit`	Top value for the scale, defaults to 2 (numeric, optional, defaults).
`replace`	Should the existing 'ratio' values be overwritten or stored in a new column (boolean, defaults to TRUE).

Examples

```
ratioDF <- normaliseRatio(W303) ## scales to 1 to 2 range, replaces original values.
ratioDF <- normaliseRatio(W303,rFactor=1.41,replace=FALSE)
# (multiplies score values by 1.41 and keeps the original values)
```

plotBed *A function to boxplot 'score' column of a BED dataframe, per unique chromosome name in the 'chrom' column. The resulting plot also highlights outliers based on the inter quartile range (IQR). The genome wide median is plotted as a pink line through the boxplots.*

Description

A function to boxplot 'score' column of a BED dataframe, per unique chromosome name in the 'chrom' column. The resulting plot also highlights outliers based on the inter quartile range (IQR). The genome wide median is plotted as a pink line through the boxplots.

Usage

```
plotBed(bed, plotting = TRUE)
```

Arguments

bed	A dataframe containing 'score' and 'chrom' columns (dataframe).
plotting	Should the plot object be sent to the default device? (boolean, defaults to TRUE).

Examples

```
plotBed(W303_S)
plotObject <- plotBed(W303_G2,plotting=FALSE)
```

plotCoverage *A function to scatterplot 'score' column of a BED dataframe plotCoverage function plots values in the 'score' column of the supplied bed dataframe as a function of chromosome coordinates. The genome wide median is plotted as a pink line.*

Description

A function to scatterplot 'score' column of a BED dataframe plotCoverage function plots values in the 'score' column of the supplied bed dataframe as a function of chromosome coordinates. The genome wide median is plotted as a pink line.

Usage

```
plotCoverage(bed, region = FALSE, plotting = TRUE)
```

Arguments

bed	A dataframe containing 'score','chrom','chromStart' and 'chromEnd' columns (dataframe).
region	Only plot for the provided region in the format 'chrI:1000-3000' (string, optional).
plotting	Should the plot object be sent to the default device? (boolean, defaults to TRUE).

Examples

```
plotCoverage(W303_G2)
plotObject <- plotCoverage(W303_S,plotting=FALSE)
```

---

plotGenome *plotGenome: plot replication profile.*

Description

plotGenome plots scatterplot/barplot/polygon of 'score' and/or 'splineSmooth' columns values by genomic coordinates, either for the whole genome or a region. It also allows annotation with additional BED-formatted data. Ggplot2 package is used for plotting.

Usage

```
plotGenome(
  ratioDFs,
  geom = "geom_point",
  ylims = c(1, 2),
  plotting = TRUE,
  genome = NULL,
  region = FALSE,
  guide = NULL,
  lines = NULL,
  circles = NULL,
  rectangles = NULL,
  pointers = NULL,
  colourLines = "#00FF00",
  colourCircles = "#FFFFFF",
  colourRectangles = "#FF0000",
  colourPointers = "#FF7F00"
)
```

Arguments

ratioDFs	A ratio dataframe or combined ratios dataframe containing 'ratio' column (dataframe).
geom	ggplot2 geom to use for plotting: "geom_point","geom_ribbon" or "geom_segment" (string, defaults to "geom_point").
ylims	A vector of two values for y axis limits - first is lowest, second is highest (numeric vector, defaults to 1 and 2)
plotting	Should the plot object be sent to the default device? (boolean, defaults to TRUE).
genome	A mask dataframe to exclude data from the ratio dataframe (dataframe, optional). The genome dataframe must contain "chrom","chromStart" and "chromEnd" columns.
region	Only plot for the provided region in the format 'chrI:1000-3000' (string, optional).
guide	A dataframe guiding the plotGenome function how to plot the data (dataframe, optional). The guide dataframe must contain the following columns: 'order' (integer) - order to plot data in, 'name.rep' (character) - replicating sample name that matches the one in the ratioDFs dataframe, 'name.nonRep' (character) - non-replicating sample name that matches the one in the ratioDFs dataframe, 'raw' (logical) - should the raw raw data be plotted? 'smooth' (logical) - should the smoothed data be plotted? 'color'(character) - R color to plot the current sample with, both raw and smoothed data.
lines	Additionally plot vertical lines from a BED formatted dataframe (dataframe, optional).
circles	Additionally plot circles on the chromosome line from a BED formatted dataframe (dataframe, optional).
rectangles	Additionally plot rectangles on the chromosome line from a BED formatted dataframe (dataframe, optional).
pointers	Additionally plot downward pointing triangles from a BED formatted dataframe (dataframe, optional).
colourLines	Colour for 'lines' data (string, defaults to green).
colourCircles	Colour for 'circles' data (string, defaults to white).
colourRectangles	Colour for 'rectangles' data (string, defaults to red).
colourPointers	Colour for 'pointers' data (string, defaults to orange).

Examples

```
plotGenome(sortSeq,geom="geom_ribbon",guide=guide,region="chrIX:250000-439885",
    lines=sacCer3[["cen"]],circles=sacCer3[["ori"]])
# plot data as polygon for the specified region of chromosome 9

plotGenome(syncSeq[["data"]],geom="geom_segment",guide=syncSeq[["guide"]],
    region="chrVII:0-1090944",genome=sacCer3[["genome"]],lines=sacCer3[["cen"]],
    circles=sacCer3[["ori"]],colourLines="black")
```

```
plotGenome(MFAseq,region='chr1:0-2848000')
# plot marker frequency analysis for H.volcanii isolate DS2
```

`plotRatio` *A function to plot a histogram of supplied ratio vector plotRatio plots histogram of values in a supplied vector using ggplot2 and highlights interval between 1 and 2 in green.*

Description

A function to plot a histogram of supplied ratio vector plotRatio plots histogram of values in a supplied vector using ggplot2 and highlights interval between 1 and 2 in green.

Usage

```
plotRatio(ratio, plotting = TRUE)
```

Arguments

`ratio`	A numeric vector containing raw or smoothed ratio values (vector).
`plotting`	Should the plot object be sent to the default device? (boolean, defaults to TRUE).

Examples

```
plotRatio(W303$ratio)
plotObject <- plotRatio(W303$ratio,plotting=FALSE)
```

`plotTrep` *A function to scatterplot 'Trep' column of a Trep dataframe plotTrep function plots values in the 'Trep' column of the supplied dataframe as a function of chromosome coordinates. The genome wide median is plotted as a pink line.*

Description

A function to scatterplot 'Trep' column of a Trep dataframe plotTrep function plots values in the 'Trep' column of the supplied dataframe as a function of chromosome coordinates. The genome wide median is plotted as a pink line.

Usage

```
plotTrep(TrepDF, region = FALSE, plotting = TRUE)
```

Arguments

TrepDF	A dataframe containing 'chrom','chromStart', 'chromEnd' and 'Trep' columns (dataframe).
region	Only plot for the provided region in the format 'chrI:1000-3000' (string, optional).
plotting	Should the plot object be sent to the default device? (boolean, defaults to TRUE).

Examples

```
plotTrep(TrepDF,region="chrI")
```

---

rmChr *A function to remove single chromosome data from a bed dataframe*

Description

A function to remove single chromosome data from a bed dataframe

Usage

```
rmChr(bed, chr)
```

Arguments

bed	A bed dataframe containing 'chrom' column (dataframe, required).
chr	Chromosome to remove (string, required).

Examples

```
bedDF <- rmChr(W303_S,"chrM") ## removes mitochondria
```

---

rmOutliers *A function to remove outliers from the "score" column of a supplied bed dataframe There are three methods: max, IQR and median. Max is used to remove 1 or more maximum values; IQR uses interquartile range to detect outliers, while median method can be used to remove data based on genome-wide median.*

Description

A function to remove outliers from the "score" column of a supplied bed dataframe There are three methods: max, IQR and median. Max is used to remove 1 or more maximum values; IQR uses interquartile range to detect outliers, while median method can be used to remove data based on genome-wide median.

Usage

```
rmOutliers(bed, method, n = 1, range = 3, loLim = 0.25, hiLim = NULL)
```

Arguments

bed	A dataframe containing 'score' column (dataframe, required).
method	Method to detect outliers: "max", "IQR" or "median" (string).
n	Number of max values to remove (integer,defaults to 1). Use with "max" method.
range	Number of IQR above the 3rd or below the 1st IQR to set the threshold (double, defaults to 3). Use with "IQR" method.
loLim	Low limit for the median method (double, defaults to 0.25).
hiLim	High limit for the median method (double).

Examples

```
bedDF <- rmOutliers(W303_S,method="max",n=2) ## removes 2 rows of data containing 3 top values
bedDF <- rmOutliers(W303_S,method="IQR",range=3) ## removes datapoints outside 3 x IQR above the 3rd
 # and below the 1st IQR.
bedDF <- rmOutliers(W303_S,method="median",loLim=0.25,hiLim=2) # removes datapoints that are lower
 # than 0.25 x genome median or above 2 x genome median.
```

---

runGUI *A function to launch Repliscope in interactive mode (Shiny app).*

Description

A function to launch Repliscope in interactive mode (Shiny app).

Usage

```
runGUI()
```

---

sacCer3 *S.cerevisiae genome information*

Description

sacCer3 meta information: chromosome sizes, centromere and replication origin positions.

Usage

```
data(sacCer3)
```

Format

List containing three dataframes

genome Chromosome information dataframe

cen Centromere information dataframe

ori Replication origin information dataframe

Source

Replication origin information: (OriDB)

References

Siow et al. (2011) NAR 40(Database issue):D682-6 (PubMed)

Examples

```
data(sacCer3)
```

`smoothRatio` *A function to smooth ratio values using cubic smoothing spline smoothRatio function splits values from 'ratio' column by chromosome and based the supplied 'groupMin' and 'split' parameters and then applies smooth.spline() function from R stats package. The supplied dataframe may contain multiple ratios, i.e. ratios produced using multiple replicating samples and/or multiple non-replicating samples. This must be reflected in 'name.rep' and 'name.nonRep' columns. In other words, different ratio dataframes may be merged using rbind() function before calling smoothRatio() function.*

Description

A function to smooth ratio values using cubic smoothing spline smoothRatio function splits values from 'ratio' column by chromosome and based the supplied 'groupMin' and 'split' parameters and then applies smooth.spline() function from R stats package. The supplied dataframe may contain multiple ratios, i.e. ratios produced using multiple replicating samples and/or multiple non-replicating samples. This must be reflected in 'name.rep' and 'name.nonRep' columns. In other words, different ratio dataframes may be merged using rbind() function before calling smoothRatio() function.

Usage

```
smoothRatio(ratioDF, groupMin = 5, splitNum = 5)
```

Arguments

ratioDF	A ratio dataframe or combined ratios dataframe containing 'ratio' column (dataframe).
groupMin	Minimum number of values required to make a group (integer, defaults to 5).
splitNum	Minimum number of adjacent bins with missing values to close current group (integer, defaults to 5).

Examples

```
ratioDF <- smoothRatio(W303norm)
```

---

sortSeq *Replication profiles for wild type and Dbf4-9myc S.cerevisiae samples*

Description

Replication profiles for wild type and Dbf4-9myc samples (T7107 and T9394 strains). The cells were stained with DNA dye and sorted based on DNA content into S or G2/M phase fractions. Extracted DNA was sequenced and mapped to sacCer3 genome. Unique reads for replicating (S) and non-replicating (G2/M) samples were calculated in 1 kb genomic bins. The ratio was created by dividing 'score' values from replicating sample by non- replicating sample 'score' values, adjusted by total number of reads. The ratio values were further adjusted by multiplying them by 1.41 and 1.402 for wild type and Dbf4-9myc samples, respectively, to put the values onto biologically relevant relative copy number scale from 1 to 2. The relative copy number values were smoothed using cubic spline and compared using z score statistics.

Usage

```
data(sortSeq)
```

Format

data frame with 22696 rows and 10 variables:

chrom short chromosome name

chromStart left chromosome coordinate

chromEnd right chromosome coordinate

name.rep replicating sample name

name.nonRep non-replicating sample name

ratio ratio value in the current bin

ratioFactor adjustment factor used for the current ratio

group Group number of the current bin

splineSmooth Smoothed ratio value

p.value Significance of ratio difference between Dbf4myc and W303 samples

Source

Dbf4myc S phase sample: SRA; Dbf4myc G2 sample: SRA; W303 S sample: SRA; W303 G2 sample: SRA

References

Natsume et al. (2013) Mol Cell 50(5):661-74 (PubMed)

Examples

```
data(sortSeq)
```

---

syncSeq *Replication profiles budding yeast arrest-release samples*

Description

Replication profiles of wild type S. cerevisiae arrest-release samples (AUY077 strain). The cells were arrested in G1 with alpha-factor followed by release using pronase. The samples were collected before the release (aFactor) and various time intervals after the release (25min,30min,35min,40min,45min,50min and 90min) Extracted DNA was sequenced and mapped to sacCer3 genome. Unique reads for replicating (post-release) and non-replicating (aFactor) samples were calculated in 1 kb genomic bins. The ratios were created by dividing 'score' values from replicating samples by non-replicating sample 'score' values, adjusted by total number of reads. The ratio values were further adjusted based on bulk genome replication (as determined by flow cytometry), to put the values onto biologically relevant relative copy number scale from 1 to 2. The relative copy number values were smoothed using cubic spline.

Usage

```
data(syncSeq)
```

Format

List containing two data frames

data syncSeq replication profiles data. Columns: **chrom** (short chromosome name), **chromStart** (left chromosome coordinate), **chromEnd** (right chromosome coordinate), **name.rep** (replicating sample name), **name.nonRep** (non-replicating sample name), **ratio** (ratio value in the current bin), **ratioFactor** (adjustment factor used for the current ratio), **group** (Group number of the current bin), **splineSmooth** (Smoothed ratio value)

guide Guide dataframe for plotting the syncSeq data **order** (Order to plot data in), **name.rep** (Name of replicating sample), **name.nonRep** (Name of non-replicating sample), **raw** (Should raw data be plotted?), **smooth** (Should smooth data be plotted?), **color** (Color to plot the profile in)

Source

GEO

References

Müller et al. (2014) NAR 42(1):e3 (PubMed)

Examples

```
data(syncSeq)
```

`TrepDF` *Trep data calculated from syncSeq[["data"]]*

Description

Trep is median replication time, expressed in minutes after release G1 arrest. It is calculated from multiple relative copy number datapoints across timeseries of a cell cycle experiment. For every genomic bin, a sigmoid function is fitted and its midpoint is reported.

Usage

```
data(TrepDF)
```

Format

data frame with 11341 rows and 5 variables:

chrom short chromosome name

chromStart left chromosome coordinate

chromEnd right chromosome coordinate

Trep calculated Trep value

TrepErr error from sigmoid function fitting

References

Müller et al. (2014) NAR 42(1):e3 (PubMed)

Examples

```
data(TrepDF)
```

`trimRatio` *A function to remove outliers from the "ratio" column of a supplied ratio dataframe trimRatio is applied to the calculated ratio of read counts from a replicating to a non-replicating samples.*

Description

A function to remove outliers from the "ratio" column of a supplied ratio dataframe trimRatio is applied to the calculated ratio of read counts from a replicating to a non-replicating samples.

Usage

```
trimRatio(ratioDF, loLim, hiLim)
```

Arguments

`ratioDF`	A ratio dataframe containing 'ratio' column (dataframe, required).
`loLim`	Low limit threshold (double, required).
`hiLim`	High limit threshold (double, required).

Examples

```
W303 <- trimRatio(W303,0.5,1.5)
```

`W303` *Sequence read coverage ratios for wild type S.cerevisiae W303*

Description

Sequence read coverage ratios for wild type sample (T7107 strain). The cells were stained with DNA dye and sorted based on DNA content into S or G2/M phase fractions. Extracted DNA was sequenced and mapped to sacCer3 genome. Unique reads for replicating (S) and non-replicating (G2/M) samples were calculated in 1 kb genomic bins. The ratio was created by dividing 'score' values from replicating sample by non- replicating sample 'score' values, adjusted by total number of reads.

Usage

```
data(W303)
```

Format

data frame with 11350 rows and 7 variables:

chrom short chromosome name

chromStart left chromosome coordinate

chromEnd right chromosome coordinate

name.rep replicating sample name

name.nonRep non-replicating sample name

ratio ratio value in the current bin

ratioFactor adjustment factor used for the current ratio

Source

S phase sample: SRA; G2 sample: SRA

References

Natsume et al. (2013) Mol Cell 50(5):661-74 (PubMed)

Examples

```
data(W303)
```

`W303norm` *Normalised sequence read coverage ratios for wild type S.cerevisiae W303*

Description

Sequence read coverage ratios for wild type sample (T7107 strain). The cells were stained with DNA dye and sorted based on DNA content into S or G2/M phase fractions. Extracted DNA was sequenced and mapped to sacCer3 genome. Unique reads for replicating (S) and non-replicating (G2/M) samples were calculated in 1 kb genomic bins. The ratio was created by dividing 'score' values from replicating sample by non- replicating sample 'score' values, adjusted by total number of reads. The ratio values were further adjusted by multiplying them by 1.41 to put the values onto biologically relevant relative copy number scale from 1 to 2.

Usage

```
data(W303norm)
```

Format

data frame with 11340 rows and 7 variables:

chrom short chromosome name

chromStart left chromosome coordinate

chromEnd right chromosome coordinate

name.rep replicating sample name

name.nonRep non-replicating sample name

ratio ratio value in the current bin

ratioFactor adjustment factor used for the current ratio

Source

S phase sample: SRA; G2 sample: SRA

References

Natsume et al. (2013) Mol Cell 50(5):661-74 (PubMed)

Examples

```
data(W303norm)
```

`W303_G2` *Sequence read coverage for wild type S.cerevisiae W303 non-replicating sample.*

Description

Sequence read coverage for wild type non-replicating sample (T7107 strain). The cells were stained with DNA dye and sorted based on DNA content into G2/M phase fraction. Extracted DNA was sequenced and mapped to sacCer3 genome. Unique reads were calculated in 1 kb genomic bins.

Usage

```
data(W303_G2)
```

Format

data frame with 11350 rows and 5 variables:

chrom short chromosome name

chromStart left chromosome coordinate

chromEnd right chromosome coordinate

name sample name

score read number in current bin

Source

SRA

References

Natsume et al. (2013) Mol Cell 50(5):661-74 (PubMed)

Examples

```
data(W303_G2)
```

`W303_S` *Sequence read coverage for wild type S.cerevisiae W303 replicating sample*

Description

Sequence read coverage for wild type replicating sample (T7107 strain). The cells were stained with DNA dye and sorted based on DNA content into S phase fraction. Extracted DNA was sequenced and mapped to sacCer3 genome. Unique reads were calculated in 1 kb genomic bins.

Usage

```
data(W303_S)
```

Format

data frame with 11820 rows and 5 variables:

chrom short chromosome name

chromStart left chromosome coordinate

chromEnd right chromosome coordinate

name sample name

score read number in current bin

Source

SRA

References

Natsume et al. (2013) Mol Cell 50(5):661-74 (PubMed)

Examples

```
data(W303_S)
```

Index