Package ‘deltaCaptureC’

May 24, 2024

**Title**  This Package Discovers Meso-scale Chromatin Remodeling from 3C Data

**Version**  1.18.0

**Description**  This package discovers meso-scale chromatin remodelling from 3C data. 3C data is local in nature. It gives interaction counts between restriction enzyme digestion fragments and a preferred 'viewpoint' region. By binning this data and using permutation testing, this package can test whether there are statistically significant changes in the interaction counts between the data from two cell types or two treatments.

**License**  MIT + file LICENSE

**Encoding**  UTF-8

**LazyData**  true

**Depends**  R (>= 3.6)

**Imports**  IRanges, GenomicRanges, SummarizedExperiment, ggplot2, DESeq2, tictoc

**Roxygen**  list(markdown = TRUE)

**RoxygenNote**  7.2.3.9000

**Suggests**  knitr, rmarkdown

**VignetteBuilder**  knitr

**biocViews**  BiologicalQuestion, StatisticalMethod

**git_url**  https://git.bioconductor.org/packages/deltaCaptureC

**git_branch**  RELEASE_3_19

**git_last_commit**  ba5ee72

**git_last_commit_date**  2024-04-30

**Repository**  Bioconductor 3.19

**Date/Publication**  2024-05-24

**Author**  Michael Shapiro [aut, cre] (<https://orcid.org/0000-0002-2769-9320>)

**Maintainer**  Michael Shapiro <michael.shapiro@crick.ac.uk>
## Contents

- `.getRunsAndTotals` ................................................. 3
- `bigBinSize` ......................................................... 3
- `binnedDeltaPlot` .................................................... 4
- `binnedDeltaSE` ....................................................... 4
- `binnedSummarizedExperiment` ......................................... 5
- `binSummarizedExperiment` ............................................ 5
- `deltaSE` ........................................................... 6
- `downshiftDFtoMatrix` ............................................... 6
- `generatePermutation` ................................................. 7
- `getDeltaSE` .......................................................... 7
- `getLopsidedness` ..................................................... 8
- `getMeanNormalizedCountsSE` ......................................... 8
- `getNormalizedCountsSE` ............................................. 9
- `getOverlapWeights` .................................................. 10
- `getPValueCutoff` ..................................................... 10
- `getRunAndLopsidednessStatistics` .................................... 11
- `getRunStatistics` .................................................... 12
- `getRunStatisticsDist` ............................................... 12
- `getRunTotals` ....................................................... 13
- `getSignificantRegions` ............................................... 13
- `getSizeFactorsDF` ................................................... 14
- `getSizeFactorsSE` ................................................... 15
- `miniDeltaSE` ......................................................... 15
- `miniSE` .......................................................... 16
- `miniSEDF` ........................................................ 16
- `numPermutations` .................................................... 16
- `plotSignificantRegions` ............................................. 17
- `plotTitle` .......................................................... 18
- `pValue` .......................................................... 18
- `rebinToMultiple` .................................................... 18
- `regionOfInterest` ..................................................... 19
- `significanceType` .................................................... 19
- `significantRegions` .................................................. 20
- `significantRegionsPlot` ............................................. 20
- `smallBins` ......................................................... 21
- `smallBinSize` ....................................................... 21
- `smallerDeltaSE` ..................................................... 22
- `smallSetOfSmallBins` ............................................... 22
- `viewpointRegion` ..................................................... 22
- `weightsExampleBins` ................................................ 23
- `weightsExampleGr` ................................................... 23

## Index

24
Description

This takes a GRanges object for binneed data and a column name designating where to find the relevant data in the mcols and returns a GRanges giving the consecutive runs of constant sign and their run totals. It is not exported.

Usage

.getRunsAndTotals(gr, colName)

Arguments

gr a GRanges object whose mcols gives the relevant binned data

colName This designates the column in mcols with the relevant data

Value

a GRanges object giving the contiguous regions and their respective sums.

bigBinSize

Big bin size

Description

The deltaCaptureC package depends on rebinning mean normalized capture C data from a small bin size to a large bin size. This gives the size of the latter. This is used in package documentation.

Usage

bigBinSize

Format

a numeric scalar
binnedDeltaPlot

Description

The data plotted here are the difference between mean normalized counts for captures in two replicates each of EScells and Neurons. The data here is binned to 1000bp and is shown in a region of 500kb up- and down-stream of the Paupar viewpoint. It is used here to illustrate functioning of the deltaCaptureC package.

Usage

binnedDeltaPlot

Format

A ggplot object

binnedDeltaSE

Binned difference of mean capture-C counts between EScells and Neurons

Description

A smaller deltaSE restriction to a region of interest around the viewpoint, here binned to the big-BinSize, 10kb.

Usage

binnedDeltaSE

Format

A RangedSummarizedExperiment with 1909 rows.
binnedSummarizedExperiment

Binned Capture-C counts of EScells and Neurons

Description

Capture-C counts of EScells and Neurons restricted to a region of interest around the Paupar viewpoint. The data have been binned to smallBins, i.e., bins of size 1000. Used in package documentation.

Usage

binnedSummarizedExperiment

Format

A RangedSummarizedExperiment.

binSummarizedExperiment

Bin a Summarized experiment into a set of bins given by a GRanges object

Description

This function takes a set of bins given by a GRanges object and a RangedSummarizedExperiment and produces a new RangedSummarizedExperiment with the bins as its rowRanges.

Usage

binSummarizedExperiment(bins, se, checkDisjoint = FALSE)

Arguments

bins a GRanges object whose ranges should be disjoint
se a RangedSummarizedExperiment
checkDisjoint = FALSE if set to true will check that the bins are disjoint

Value

a RangedSummarizedExperiment

Examples

binnedSummarizedExperiment = binSummarizedExperiment(smallSetOfSmallBins,smallerDeltaSE)
deltaSE  
\textit{Difference of mean capture-C counts between EScells and Neurons}

**Description**

This gives the difference in mean normalized capture C counts between two replicates each of EScells and Neurons. It is the underlying kind of data on which deltaCaptureC operates and is here to support documentation.

**Usage**

deltaSE

**Format**

A RangedSummarizedExperiment with 5709 rows.

downshiftDFtoMatrix  
\textit{Downshift from DF to matrix}

**Description**

This function takes a data.frame with chr, start, end and numerical data and turns it into a matrix with row names chr:start-end

**Usage**

downshiftDFtoMatrix(df)

**Arguments**

df  
This is a data frame whose first three columns are chr, start and end and whose remaining columns are numerical data

**Value**

A matrix of numerical data

**Examples**

m = downshiftDFtoMatrix(miniSEDF)
generatePermutation  

Generate permutation for permutation testing

Description
This function takes a set of row ranges and an inner region and generates a permutation which is symmetric on the inner region and arbitrary on the remainder.

Usage
```r
generatePermutation(gr, innerRegion)
```

Arguments
- `gr` a GRanges object which should be ordered
- `innerRegion` a GRanges object which should be a single interval

Value
- a permutation of 1:length(gr)

Examples
```r
permutations = generatePermutation(smallBins,viewpointRegion)
```

getDeltaSE  

Make delta summarized experiment:

Description
This function takes a SummarizedExperiment with count data and produces a SummarizedExperiment of the delta track. There should exactly two values for treatment, i.e., byTreatment.

Usage
```r
getDeltaSE(countsSE, byTreatment = "treatment")
```

Arguments
- `countsSE` A summarized experiment with assay counts and optionally assay normalized counts
- `byTreatment` = 'treatment' Allows for specifying some other condition than 'treatment'

Value
A summarized experiment with a single assay consisting of a single column, the delta mean normalized counts.
Examples

```r
aSmallDeltaSE = getDeltaSE(miniSE)
```

---

**getLopsidedness**  
*Get the lopsidedness statistic*

---

**Description**

This function looks at the sidedness around the viewpoint and returns the absolute value of the difference between the sum of the values before and after the viewpoint inside the viewpoint region.

**Usage**

```r
getLopsidedness(se, viewpointRegion, colName = "delta")
```

**Arguments**

- `se`: a `SummarizedExperiment` giving the delta or permuted delta
- `viewpointRegion`: the region around the viewpoint in which to investigate lopsidedness
- `colName`: defaults to 'delta'

**Value**

the lopsidedness around the viewpointMid in the viewpointRegion

**Examples**

```r
lopsidedness = getLopsidedness(binnedDeltaSE,viewpointRegion)
```

---

**getMeanNormalizedCountsSE**

*Make mean treatment summarized experiment:*

---

**Description**

Get the mean normalized counts for each treatment

**Usage**

```r
getMeanNormalizedCountsSE(countsSE, byTreatment = "treatment")
```
getNormalizedCountsSE

Arguments

- countsSE: A SummarizedExperiment containing an assay ‘counts’ and optionally an assay ‘normalizedCounts’
- byTreatment = ‘treatment’ This gives the column of colData to use for taking averages

Details

This function takes a SummarizedExperiment. It looks for an assay called normalizedCounts. If this assay is missing, it creates it by normalizing using the size factors. By default, it takes the mean for each value of colData$treatment

Value

A SummarizedExperiment giving mean normalized counts for each value of byTreatment

Examples

```r
meanNormalizedCountSE = getMeanNormalizedCountsSE(miniSE)
```

getNormalizedCountsSE  Get normalized counts

Description

This function takes a SummarizedExperiment giving the counts for each replicate of the two treatments and computes and affixes an assay giving the normalized version of these counts.

Usage

getNormalizedCountsSE(se)

Arguments

- se: A SummarizedExperiment with an assay called counts giving the raw counts for each replicate of the two treatments.

Value

A SummarizedExperiment including a an assay of the normalized counts called normalizedCounts.

Examples

```r
miniSENormalized = getNormalizedCountsSE(miniSE)
```
getOverlapWeights

Get the binning factors for one set of GRanges into another

Description
This function takes two GRanges, one representing a set of bins and the other representing data to be pro-rated over those bins and returns a data frame giving the overlaps, various widths and the fractions for pro-rating scores.

Usage
getOverlapWeights(bins, gr, checkDisjoint = FALSE)

Arguments
- bins: a set of GRanges to be used for binning data.
- gr: the GRanges of the data to be binned.
- checkDisjoint = FALSE if this is TRUE it will check to see that the ranges in each of bins and gr are disjoint.

Value
A data frame giving index pairs for the intersections, widths of the intersections and the fraction of each gr range meeting each bin.

Examples
overlapWeights = getOverlapWeights(weightsExampleBins, weightsExampleGr)

getPValueCutoff

This function returns the significance levels for min, max, "abs" and lopsidedness.

Description
Given an Nx4 matrix with columns 'min','max','abs' and 'lopsidedness', this function returns the cutoff levels for a given pValue.

Usage
getPValueCutoff(runStats, p = .05)

Arguments
- runStats: a matrix with columns 'min','max','abs' and 'lopsidedness'.
- p = .05 the desired p-value.
getRunAndLopsidednessStatistics

Value

a vector with cutoff values

Examples

dimnames = list(c(),c('min','max','abs','lopsidedness'))
m = 10 * (matrix(runif(400),ncol=4,dimnames=dimnames) - 0.5)
cutoffs = getPValueCutoff(m,.05)

Description

Get the distribution of run and lopsidedness statistics

Usage

getRunAndLopsidednessStatistics(
  scrambledDeltas,
  viewpointRegion,
  colName = "delta"
)

Arguments

scrambledDeltas
  a list of rebinned (i.e., to large bin size) of scrambled deltas

viewpointRegion
  a GRanges object giving the region that is reserved for lopsidedness

colName = 'delta'

Value

a Nx4 matrix giving the min, max, max(abs(min),abs(max)) and lopsidedness for the run totals in the list of scrambled deltas.
getRunStatistics

Description
This function is called by getRunsStatisticsDist on the individual elements of a list of scrambled runs.

Usage
getRunStatistics(runTotals)

Arguments
runTotals is a GRanges object giving the consecutive runs and their totals.

Value
a vector of the min, max and absolute value of the min and max for the run totals.

getRunStatisticsDist

Description
This takes a list of (scrambled) runs and returns their run statistics

Usage
getRunStatisticsDist(runTotalsList)

Arguments
runTotalsList this is a list whose members are GRanges objects giving the consecutive runs and their totals

Value
a Nx3 matrix giving the min, max and max(abs(min),abs(max)) run totals
getRunTotals

Description

This function finds the runs of consecutive ranges in which the sign of the data does not change. It returns a GRanges object containing the contiguous ranges and the weighted sum of data in each.

Usage

getRunTotals(se, innerRegion, colName = "delta")

Arguments

se                a SummarizedExperiment whose first assay has a column named colName. Typically this will be a one-column matrix with delta.
innerRegion       a Granges object defining the region surrounding the viewpoint to be excluded from run total calculations
colName           defaults to 'delta'

Value

a GRanges object giving the contiguous regions and their respective sums

Examples

runTotals = getRunTotals(binnedDeltaSE,viewpointRegion)

getSignificantRegions

Get the significant regions from delta data

Description

This function takes delta data as a SummarizedExperiment and required ancillary data and returns a GenomicRanges object whose mcols indicate the significant regions.

Usage

getSignificantRegions(
    deltaSE,
    regionOfInterest,
    viewpointRegion,
    smallBinSize,
    bigBinSize,
    numPermutations = 1000,
    pValue = 0.05
)
getSizeFactorsDF

Get the size factors for count normalization

Arguments

- **deltaSE**: a ranged summarized experiment with a one-column assay giving the delta mean count
- **regionOfInterest**: a GenomicRanges object specifying the region of interest
- **viewpointRegion**: the region withheld from arbitrary permutation
- **smallBinSize**: size to bin original data to for permutation
- **bigBinSize**: size to bin data to for significance testing. Must be a multiple of smallBinSize
- **numPermutations**: = 1000 the number of permutations to be used for permutation testing
- **pValue**: the desired significance level

Value

- A GRanges object giving the bigBin binning of region of interest whose mcols gives the values of delta and logicals telling whether the bin is in the viewpoint region and whether it rises to statistical significance

Description

This function takes a data frame giving chr, start, end and count for experimental replicates and returns the size factors for each of the replicates for use in normalization

Usage

getSizeFactorsDF(countsDF)

Arguments

- **countsDF**: A data frame whose first three columns are chr, start and end, and whose remaining columns are count data for experimental replicates

Value

The size factors for the columns of countsDF

Examples

sf = getSizeFactorsDF(miniSEDF)
**getDescriptionSE**

Get the size factors for SummarizedExperiment

**Description**

This function takes a SummarizedExperiment with an assay counts and returns this object with a column sizeFactors added to its colData

**Usage**

getDescriptionSE(se)

**Arguments**

se

A SummarizedExperiment with an assay counts

**Value**

The same SummarizedExperiment with an additional column in its colData giving the size factors for counts

**Examples**

miniSEWithSizeFactors = getDescriptionSE(miniSE)

---

**miniDeltaSE**

Difference of mean capture-C counts between ESCells and Neurons

**Description**

A smaller deltaSE restriction to a region of interest around the viewpoint.

**Usage**

miniDeltaSE

**Format**

A RangedSummarizedExperiment with 1909 rows.
miniSE

**Description**
Capture-C counts of EScells and Neurons restricted to a region of interest around the Paupar viewpoint. Used in package documentation.

**Usage**

```
miniSE
```

**Format**
A RangedSummarizedExperiment with 1909 rows.

miniSEDF

**Description**
Capture-C counts of EScells and Neurons restricted to a region of interest around the Paupar viewpoint. This contains the same information as miniSE, but here packaged as a data frame. Used in package documentation.

**Usage**

```
miniSEDF
```

**Format**
A data frame with 1909 rows.

numPermutations

**Description**
Number of permutations used in example permutation testing in the example in the documentation.

**Usage**

```
umPermutations
```

**Format**
A numeric scalar
plotSignificantRegions

This produces a plot of the region of interest showing regions of significance.

Description

This function takes a input the GRanges object produced by getSignificant regions and produces a ggplot of significant features

Usage

plotSignificantRegions(
    significantRegions,
    significanceType = "abs",
    title = "Significant Regions",
    xLabel = "viewpoint",
    legend = TRUE
)

Arguments

significantRegions
  a GRanges object as produced by getSignificantRegions

significanceType
  = 'abs' a variable indicating whether to plot significance according to min, max or abs.

title
  a title for the plot

xLabel
  = 'viewpoint' supplies an xlabel

legend
  = TRUE whether or not to show the legend

Value

a ggplot object

Examples

plotOfSignificantRegions = plotSignificantRegions(significantRegions)
plotTitle  Title for delta capture-C plot

Description
A title for the illustrative plot in the documentation

Usage
plotTitle

Format
A character scalar

pValue  P-value

Description
P-value used in the illustrative example of discovering statistically significant chromatin remodeling. Used in package documentation.

Usage
pValue

Format
A numeric scalar

rebinToMultiple  Rebin a SummarizedExperiment to a multiple of its bin width

Description
This is a faster way of rebinning when the old bins are consecutive and constant width and the new bins are to be a multiple of that width

Usage
rebinToMultiple(se, multiple, deleteShort = FALSE)
Arguments

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>se</td>
<td>a RangedSummarizedExperiment to be rebinned</td>
</tr>
<tr>
<td>multiple</td>
<td>the factor by which to fatten the bins</td>
</tr>
<tr>
<td>deleteShort</td>
<td>= FALSE when set to true if the final bin is short it will be deleted</td>
</tr>
</tbody>
</table>

Value

a RangedSummarizedExperiment

Examples

rebinnedSummarizedExperiment = rebinToMultiple(binnedSummarizedExperiment, 10)

regionOfInterest

Region of interest surrounding the viewpoint

Description

This is the region 500kb up- and down-stream of the Paupar viewpoint, and defines the region on which we will look for statistically significant chromatin remodeling. This is used in package documentation.

Usage

regionOfInterest

Format

A GRanges object with one segment.

significanceType

Type for testing significance

Description

Significance testing can be carried out by maximum, minimum or maximum absolute value. This has the value abs thus choosing the last of these. This is used in package documentation.

Usage

significanceType

Format

A scalar character
Description

A GRanges object giving the value of delta in each bin and the significance or lack thereof according to the various statistical tests. This is used in package documentation.

Usage

significantRegions

Format

A GRanges object

Description

A ggplot object show the values of delta in the region of interest and the significant sub-regions. This is used for documentation purposes.

Usage

significantRegionsPlot

Format

a ggplot object
**smallBins**

<table>
<thead>
<tr>
<th>smallBins</th>
<th>Small Bins</th>
</tr>
</thead>
</table>

**Description**

The deltaCaptureC package depends on rebinning mean normalized capture C data from a small bin size to a large bin size. These are the small bins. They have a bin size of 1000 bp, i.e., smallBinSize, and span the region of interest. This is used in package documentation.

**Usage**

smallBins

**Format**

a GRanges object

---

**smallBinSize**

<table>
<thead>
<tr>
<th>smallBinSize</th>
<th>Small Bin Size</th>
</tr>
</thead>
</table>

**Description**

The deltaCaptureC package depends on rebinning mean normalized capture C data from a small bin size to a large bin size. This gives the size of the former. This is used in package documentation.

**Usage**

smallBinSize

**Format**

a numeric scalar
smallerDeltaSE  

**Description**

This is a subset of miniDeltaSE. It’s here so that the example for binSummarizedExperiment will run more quickly.

**Usage**

smallerDeltaSE

**Format**

A RangedSummarizedExperiment.

smallSetOfSmallBins  

**Description**

This is a subset of the bins in smallBins and is here so that the example for binSummarizedExperiment will run more quickly.

**Usage**

smallSetOfSmallBins

**Format**

a GRanges object

viewpointRegion  

**Description**

This is the region 50kb up- and down-stream of the Paupar viewpoint. This is the region in which counts strongly depend on distance from the viewpoint and will be treated distinctly from the region of interest.

**Usage**

viewpointRegion
weightsExampleBins

Format

A GRanges object with one segment.

weightsExampleBins  Weights example bins

Description

This GRanges object is here to support the example for the function getOverlapWeights().

Usage

weightsExampleBins

Format

a GRanges object

weightsExampleGr  Weights example

Description

This GRanges object is here to support the example for the function getOverlapWeights().

Usage

weightsExampleGr

Format

a GRanges object
Index

* datasets
  bigBinSize, 3
  binnedDeltaPlot, 4
  binnedDeltaSE, 4
  binnedSummarizedExperiment, 5
  deltaSE, 6
  miniDeltaSE, 15
  miniSE, 16
  miniSEDF, 16
  numPermutations, 16
  pValue, 18
  regionOfInterest, 19
  significanceType, 19
  significantRegions, 20
  significantRegionsPlot, 20
  smallBins, 21
  smallBinSize, 21
  smallerDeltaSE, 22
  smallSetOfSmallBins, 22
  viewpointRegion, 22
  weightsExampleBins, 23
  weightsExampleGr, 23

* dataset
  plotTitle, 18
  .getRunsAndTotals, 3

  bigBinSize, 3
  binnedDeltaPlot, 4
  binnedDeltaSE, 4
  binnedSummarizedExperiment, 5
  binSummarizedExperiment, 5
  deltaSE, 6
  downshiftDFtoMatrix, 6

  generatePermutation, 7
  getDeltaSE, 7
  getLopsidedness, 8
  getMeanNormalizedCountsSE, 8
  getNormalizedCountsSE, 9

  getOverlapWeights, 10
  getPValueCutoff, 10
  getRunAndLopsidednessStatistics, 11
  getRunStatistics, 12
  getRunStatisticsDist, 12
  getRunTotals, 13
  getSignificantRegions, 13
  getSizeFactorsDF, 14
  getSizeFactorsSE, 15

  miniDeltaSE, 15
  miniSE, 16
  miniSEDF, 16

  numPermutations, 16

  plotSignificantRegions, 17
  plotTitle, 18
  pValue, 18

  rebinToMultiple, 18
  regionOfInterest, 19

  significanceType, 19
  significantRegions, 20
  significantRegionsPlot, 20
  smallBins, 21
  smallBinSize, 21
  smallerDeltaSE, 22
  smallSetOfSmallBins, 22

  viewpointRegion, 22

  weightsExampleBins, 23
  weightsExampleGr, 23