Package 'CHARGE'

October 16, 2019

Type Package

Title CHARGE: CHromosome Assessment in R from Gene Expression data

Version 1.4.0

Date 2017-10-11

Author Benjamin Mayne

Maintainer Benjamin Mayne <benjamin.mayne@sahmri.com>

Depends R (>= 3.5), GenomicRanges, methods

Imports SummarizedExperiment, FactoMineR, factoextra, IRanges, graphics, modes, parallel, plyr, cluster, diptest, stats, matrixStats

Suggests roxygen2, EnsDb.Hsapiens.v86

biocViews GeneExpression, Clustering

Description Identifies genomic duplications or deletions from gene expression data.

License GPL-2

Encoding UTF-8

LazyData true

RoxygenNote 6.0.1

Collate 'bimodalTest.R' 'CHARGE.R' 'clusterExpr.R' 'cvExpr.R' 'exprFinder.R' 'pcaExpr.R' 'plotcvExpr.R'

git_url https://git.bioconductor.org/packages/CHARGE

git_branch RELEASE_3_9

git_last_commit f759c67

git_last_commit_date 2019-05-02

Date/Publication 2019-10-15

R topics documented:

`bimodalTest` *bimodalTest*

Description

Performs a bimodal test and calcualtes Hartigan's statistic and p-value over a genomic region of interest using gene expression data set, the output from cvExpr.

Usage

```
bimodalTest(se, cvExpr, threshold = NULL)
```

Arguments

`se`	A SummarizedExperiment containing the normalised gene expression data.
`cvExpr`	The output from cvExpr.
`threshold`	Optional. The quantile threshold of genes to be used for clustering analaysis. Default is NULL.

Details

Performs a bimodal test and calculates Hartigan's dip test statistic for unimodality for a given gene expression data set. A bimodality coefficient value > 5/9 suggests bimodality and the closer the bimodality ratio is to 1, the more evenly distrubted the data set. The dip statistic and p-value can be used to determine if the region of interest is statistically significant.

The second part of the function returns the Z score means which can be used to visualise the denisty or distribution of the samples.

Value

Returns a list containing the output from the bimodal test.

Author(s)

Benjamin Mayne

Examples

```
library(GenomicRanges)
data(datExprs)
chr21 <- GRanges("21:1-46709983")
cvExpr.out <- cvExpr(se = datExprs, region = chr21)
bimodalTest.out <- bimodalTest(se = datExprs, cvExpr = cvExpr.out, threshold = "25%")
```

`CHARGE` *CHARGE*

Description

CHARGE

clusterExpr *clusterExpr*

Description

Performs a partitioning clustering analysis in order to predict which samples have an enrichment for a genomic region of interest.

Usage

```
clusterExpr(se, cvExpr, threshold = NULL)
```

Arguments

se	A SummarizedExperiment containing the normalised gene expression data.
cvExpr	The output from cvExpr.
threshold	Optional. The quantile threshold of genes to be used for clustering analaysis. Default is NULL.

Details

Performs a partitioning clustering to predict which samples have a genomic duplication or deletion of a genomic region of interest. Samples are labelled Hyperploidy or Hypoploidy with respect to one another which are arbitrary labels referring to an enrichment or loss of genomic region.

Value

Returns a SummarizedExperiment containing the original inputted se, but where an additional column labelled Ploidy has been added into the meta data containing the classification of each sample.

Author(s)

Benjamin Mayne

Examples

```
library(SummarizedExperiment)
library(GenomicRanges)
data(datExprs)
chr21 <- GRanges("21:1-46709983")
cvExpr.out <- cvExpr(se = datExprs, region = chr21)
datExprs <- clusterExpr(se = datExprs, cvExpr = cvExpr.out, threshold = "25%")
colData(datExprs)$Ploidy
```

cvExpr *cvExpr*

Description

Calculates the coefficient of variation for each gene within a defined genomic region.

Usage

```
cvExpr(se, region)
```

Arguments

se	A SummarizedExperiment containing the normalised gene expression data.
region	A GRanges object containing the genomic location of the region of interest. This can either be an entire length or the subset of a chromosome.

Details

Calculates the coefficient of variation (CV) for each gene with a genomic region of interest. The CV Values can be used to determine which genes are not critical for appropiate clustering and can be filtered out prior to clustering.

Value

Returns a list containing the CV of each gene and the the quantile threshold of the data.

Author(s)

Benjamin Mayne

Examples

```
library(GenomicRanges)
data(datExprs)
chr21 <- GRanges("21:1-46709983")
cvExpr.out <- cvExpr(se = datExprs, region = chr21)
```

datExprs *A SummarizedExperiment from a publicly available RNA-seq data set, where the samples were sequenced on a Illumina HiSeq 2000. The data set is GSE55504 and contains 16 primary fibroblast samples where 8 are euploid for chromosome 21 and the other 8 are trisomy.*

Description

A RangedSummarizedExperiment containing normalised RNA-seq data from the data set GSE55504 which contains fibroblast samples from patients with and without Down Syndrome.

Usage

```
data(datExprs)
```

Format

RangedSummarizedExperiment

Details

- datExprs A RangedSummarizedExperiment containing normalised RNA-seq data from the data set GSE55504 which contains fibroblast samples from patients with and without Down Syndrome.

This dataset 16 fibroblast samples from patients with Down syndrome (8 samples) and are euploid for chromosome 21 (8 samples).

Value

RangedSummarizedExperiment

`exprFinder` *exprFinder*

Description

Performs a bimodality test at multiple defined bin sizes across the genome using a sliding window approach.

Usage

```
exprFinder(se, ranges, binWidth, binStep, threshold = NULL, threads = 1)
```

Arguments

`se`	A SummarizedExperiment containing the nromalised gene expression data.
`ranges`	A GRanges object containing the genomic regions to scan.
`binWidth`	The length of each bin.
`binStep`	The distance the bin will slide.
`threshold`	Optional. The quantile threshold of expression variation of genes to be used at each bin. Default is NULL.
`threads`	Total number of threads to be used. Default is 1.

Details

Uses a sliding window approach to scan over a defined genomic region. It automatically performs a bimodal test and calculates Hartigan's dip test statistic for unimodality and returns a data frame listing each bin and the statistical likelihood of a duplication or deletion.

Value

Returns a data frame containing the genomic locations of each bin and bimodality statistics.

Author(s)

Benjamin Mayne

Examples

```
library(GenomicRanges)
library(EnsDb.Hsapiens.v86)
data(datExprs)
chr21 <- GRanges("21:1-46709983")
chrLengths <- GRanges(seqinfo(EnsDb.Hsapiens.v86)[c("21", "22", "Y")])
exprFinder.out <- exprFinder(se = datExprs, ranges = chrLengths,
binWidth = 1e+9, binStep = 1e+9, threshold = "25%")
```

---

pcaExpr *pcaExprs*

Description

Creates a PCA plot using genes within a defined genomic region.

Usage

```
pcaExpr(se, cvExpr, threshold = NULL)
```

Arguments

se	A SummarizedExperiment containing the normalised gene expression data and the clustering output from clusterExpr.
cvExpr	The output from cvExpr.
threshold	Optional. The quantile threshold of genes to be used for clustering analaysis. Default is NULL.

Details

Performs a principle component analysis for a given gene expression data using only genes within a defined region.

Value

Returns a PCA plot showing the seperation of samples that were labelled hyperploidy or hypoploidy in clusterExpr.

Author(s)

Benjamin Mayne

Examples

```
library(GenomicRanges)
data(datExprs)
chr21 <- GRanges("21:1-46709983")
cvExpr.out <- cvExpr(se = datExprs, region = chr21)
datExprs <- clusterExpr(se = datExprs, cvExpr = cvExpr.out, threshold = "25%")
pcaExpr(se = datExprs, cvExpr = cvExpr.out, threshold = "25%")
```

plotcvExpr *plotcvExpr*

Description

Plots the coefficient of variation or expression variation for each gene over a defined genomic region.

Usage

```
plotcvExpr(cvExpr)
```

Arguments

cvExpr	The output from cvExpr function.

Details

Generates a bar plot showing the coefficient of variation or expression variation for each gene on the Y axis. The red, blue, green and gold horizontal lines show the 0

Value

Returns a barplot showing the CV for each gene identifier over the region of interest.

Author(s)

Benjamin Mayne

Examples

```
library(GenomicRanges)
data(datExprs)
chr21 <- GRanges("21:1-46709983")
cvExpr.out <- cvExpr(se = datExprs, region = chr21)
plotcvExpr(cvExpr = cvExpr.out)
```

Index