Package ‘partCNV’

April 4, 2024

Type Package

Title Infer locally aneuploid cells using single cell RNA-seq data

Version 1.0.0

Description This package uses a statistical framework for rapid and accurate detection of aneu-
ploid cells with local copy number deletion or amplification. Our method uses an EM algo-

rithm with mixtures of Poisson distributions while incorporating cytogenetics informa-
tion (e.g., regional deletion or amplification) to guide the classification (partCNV). When appli-
cable, we further improve the accuracy by integrating a Hidden Markov Model for feature selec-
tion (partCNVH).

Imports stats, data.table, depmixS4, Seurat, SingleCellExperiment, AnnotationHub, magrittr, GenomicRanges

Suggests BiocStyle, rmarkdown, knitr, IRanges, testthat (>= 3.0.0)

Dependents R (>= 4.2.0)

VignetteBuilder knitr

License GPL-2

Encoding UTF-8

RoxygenNote 7.2.3

biocViews Software, CopyNumberVariation, HiddenMarkovModel, SingleCell, Classification

Config/testthat/edition 3

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

git_branch RELEASE_3_18

git_last_commit 4075147

git_last_commit_date 2023-10-24

Repository Bioconductor 3.18

Date/Publication 2024-04-03

Author Ziyi Li [aut, cre, ctb], Ruoxing Li [ctb]

Maintainer Ziyi Li <zli16@mdanderson.org>
GetCytoLocation

R topics documented:

GetCytoLocation .................................................. 2
GetExprCountCyto .................................................... 3
Hg38_gtf ................................................................. 4
NormalizeCounts ....................................................... 4
partCNV ................................................................. 5
partCNVH ................................................................. 6
SimData ................................................................. 7
SimDataSce ............................................................. 7

Index 8

GetCytoLocation Get exact location of the interested cytogenetics feature

Description
This function helps you identify the location of the cytogenetics feature. For example, if the region of interest is chr20(q11.1-q13.1), this function greps the start and end location of this region. Additionally, you can just put in "chr20", and it provides you all the available cytogenetics locations on chromosome 20. It also report the number of genes within the region. If the number of genes is too few, we recommend to include neighboring regions to provide more stable results.

Usage
GetCytoLocation(cyto_feature = NULL, chr = NULL, start = NULL, end = NULL)

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>cyto_feature</td>
<td>the cytogenetics location you are interested. It can be of two format: chr20(q11.1-q13.1) or chr20. For the first format, the start and end regions need to be separated by &quot;.&quot;. If you are interested in one region for example, chr20(q11.1), put it as chr20(q11.1-11.1). For the second format, all the available regions will be printed for selection.</td>
</tr>
<tr>
<td>chr</td>
<td>chromosome location of the interested region. This is only used when cyto_feature is null.</td>
</tr>
<tr>
<td>start</td>
<td>starting location of the interested region. This is only used when cyto_feature is null.</td>
</tr>
<tr>
<td>end</td>
<td>ending location of the interested region. This is only used when cyto_feature is null.</td>
</tr>
</tbody>
</table>

Value
If the first format of cyto_feature is provided, the starting and ending location as well as the number of genes overlapped with be provided. If the second format of cyto_feature is provided, all the cytogenetics locations will be displayed for review. If the region location (chr, start, end) is provided, the number of genes overlapped will be the output.
GetExprCountCyto

Examples

### example 1
GetCytoLocation(cyto_feature = "chr20(q11.1-q13.1)"

### example 2
GetCytoLocation(cyto_feature = "chr20"

### example 3
GetCytoLocation(chr = "chr20", start = 25600000, end = 49800000)

GetExprCountCyto

Get normalized gene expression counts for selected genes

Description

This function helps normalize the gene expression count matrix if needed and select the genes that are located in the interested region. This procedure happens after applying GetCytoLocation().

Usage

GetExprCountCyto(
  cytoloc_output,
  Counts = NULL,
  normalization = TRUE,
  qt_cutoff = 0.99
)

Arguments

cytoloc_output  The output from the function GetCytoLocation(). The function needs to be run with a complete cytogenetics feature input, e.g., chr20(q11.1-11.1), or providing the chr/start/end location of the interested region.

Counts  The single cell expression matrix for the whole genome of the sample. Rows are genes and columns are cell IDs.

normalization  Specify whether the data need to be normalized. Default is TRUE.

qt_cutoff  A quantile cut-off to remove genes that are almost all zeros. If the cut-off is 0.99, then all the genes expressed in less than 0.01 percent of cells will be eliminated for further analysis.

Value

A list with normalized and ordered gene expression for the interested cytogenetics region.

Examples

res <- GetCytoLocation(cyto_feature = "chr20(q11.1-q13.1)"
data(SimData)
GetExprCountCyto(cytoloc_output = res, Counts = as.matrix(SimData), normalization = TRUE, qt_cutoff = 0.99)
**Description**

Gene location data for Hg38 genome

**Usage**

```r
data(Hg38_gtf)
```

**Format**

An object of class "data frame"

**Value**

Gene location data for Hg38 genome

**Source**

Gencode Archive

**References**


**Examples**

```r
data(Hg38_gtf)
head(Hg38_gtf)
```

---

**NormalizeCounts**

*Extract and normalize gene expression counts for a SingleCellExperiment object*

**Description**

This function helps normalize the gene expression count matrix for a SingleCellExperiment object.

**Usage**

```r
NormalizeCounts(obj, scale_factor = 10000)
```
Arguments

\texttt{obj} \quad The\ SingleCellExperiment\ object.

\texttt{scale\_factor} \quad Feature\ counts\ for\ each\ cell\ are\ divided\ by\ the\ total\ counts\ for\ that\ cell\ and
multiplied\ by\ the\ scale.factor,\ and\ then\ natural-log\ transformed\ using\ \log1p.

Value

A\ normalized\ gene\ expression\ counts\ matrix.

Examples

data(SimDataSce)
counts_mat <- NormalizeCounts(SimDataSce)

partCNV \quad \textit{Infer\ cells\ that\ are\ locally\ aneuploid\ using\ partCNV}

Description

This\ function\ uses\ EM\ algorithm\ to\ cluster\ the\ cells\ with\ a\ Poisson\ Mixture\ model.\ Cells\ will\ be
clustered\ into\ two\ groups,\ locally\ aneuploid\ (status\ =\ 1)\ and\ diploid\ (status\ =\ 0).

Usage

partCNV(int_counts, cyto_type, cyto_p, tau = 0.1, maxniter = 1000)

Arguments

\texttt{int_counts} \quad Normalized\ gene\ expression\ counts\ for\ the\ genes\ in\ the\ interested\ region,\ e.g.,
the\ ProcessedCount\ variable\ from\ the\ output\ of\ GetExprCountCyto().

\texttt{cyto\_type} \quad The\ type\ of\ the\ cytogenetics\ alteration.\ It\ can\ only\ be\ "del"\ or\ "amp"

\texttt{cyto\_p} \quad The\ percentage\ of\ cells\ with\ the\ cytogenetics\ alteration,\ e.g.,\ 0.2.

\texttt{tau} \quad The\ variance\ of\ the\ prior\ information.\ Default\ is\ 0.1.\ If\ you\ have\ less\ confidence,\ specify\ a\ larger\ tau,\ e.g.,\ 10.

\texttt{maxniter} \quad The\ maximum\ number\ of\ iterations\ of\ the\ EM\ algorithm.

Value

A\ vector\ with\ the\ cell\ status\ inferred\ by\ the\ method,\ 1\ is\ aneuploid\ and\ 0\ is\ diploid.
Examples

```r
# example 1

cytoloc <- GetCytoLocation(cyto_feature = "chr20(q11.1-q13.1)"

data(SimData)

exprout <- GetExprCountCyto(cytoloc_output = cytoloc, Counts = as.matrix(SimData), normalization = TRUE, qt_cutoff = 0.99)

status <- partCNVH(int_counts = exprout$ProcessedCount, cyto_type = "del", cyto_p = 0.2)
```

---

**partCNVH**

**Infer cells that are locally aneuploid using partCNVH**

**Description**

This function uses EM algorithm to cluster the cells with a Poisson Mixture model in the first step. With the results, it applies hidden markov model to improve feature selection. After that, another round of EM algorithm is applied to obtain the final cell status. Cells will be grouped into two groups, locally aneuploid (status = 1) and diploid (status = 0).

**Usage**

```
partCNVH(int_counts, cyto_type, cyto_p, tau = 0.1, maxniter = 1000, navg = 50)
```

**Arguments**

- **int_counts**: Normalized gene expression counts for the genes in the interested region, e.g., the ProcessedCount variable from the output of GetExprCountCyto().
- **cyto_type**: The type of the cytogenetics alteration. It can only be "del" or "amp"
- **cyto_p**: The percentage of cells with the cytogenetics alteration, e.g., 0.2.
- **tau**: The variance of the prior information. Default is 0.1. If you have less confidence, specify a larger tau, e.g., 10.
- **maxniter**: The maximum number of iterations of the EM algorithm.
- **navg**: Number of genes used for rolling average.

**Value**

A vector with the cell status inferred by the method, 1 is aneuploid and 0 is diploid.

**Examples**

```r

cytoloc <- GetCytoLocation(cyto_feature = "chr20(q11.1-q13.1)"

data(SimData)

exprout <- GetExprCountCyto(cytoloc_output = cytoloc, Counts = as.matrix(SimData), normalization = TRUE, qt_cutoff = 0.99)

status <- partCNVH(int_counts = exprout$ProcessedCount, cyto_type = "del", cyto_p = 0.2, navg = 50)
```
SimData

Simulation data to exemplify the usage of the method

Description
Simulation data to exemplify the usage of the method

Usage
data(SimData)

Format
An object of class "data frame"

Value
Simulation data to exemplify the usage of the method

Examples
data(SimData)
dim(SimData)

SimDataSce

Simulation SingleCellExperiment object to exemplify the usage of the method

Description
Simulation SingleCellExperiment object to exemplify the usage of the method

Usage
data(SimDataSce)

Format
A SingleCellExperiment object

Value
Simulation SingleCellExperiment object to exemplify the usage of the method

Examples
data(SimDataSce)
Index

* datasets
  Hg38.gtf, 4
  SimData, 7
  SimDataSce, 7

GetCytoLocation, 2
GetExprCountCyto, 3

Hg38.gtf, 4

NormalizeCounts, 4

partCNV, 5
partCNVH, 6

SimData, 7
SimDataSce, 7