Package ‘atacInferCnv’

June 4, 2026

Type Package

Title Call CNV from single cell ATAC-seq data based on InferCNV adaptation

Version 1.1.0

Description The package prepares input scATAC-seq data and adapts for copy number variance profiling with InferCNV package usage. It has also various paramters to control the analysis (e.g. external normal reference usage, meta-cells, bin size, etc) and custom plot visualizations.

License GPL-3 + file LICENSE

Encoding UTF-8

Depends R (>= 4.6.0),utils

Imports infercnv (>= 1.3.1), Signac, Seurat, GenomicRanges, GenomeInfoDb, S4Vectors, config, stringr, ggplot2, SummarizedExperiment, SingleCellExperiment, Rcpp

biocViews Epigenetics, Sequencing, CopyNumberVariation, SingleCell,ImmunoOncology

VignetteBuilder knitr

URL `https://github.com/kokonech/atacInferCNV`

BugReports `https://github.com/kokonech/atacInferCNV/issues`

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

RoxygenNote 7.3.3

LinkingTo Rcpp,RcppEigen

Config/testthat/edition 3

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

git_branch devel

git_last_commit 0347acd

git_last_commit_date 2026-04-28

Repository Bioconductor 3.24

Date/Publication 2026-06-04

Author Konstantin Okonechnikov [aut, cre] (ORCID: <`https://orcid.org/0000-0002-3409-2340`>), Supat Thongjuea [aut, fnd]

Maintainer Konstantin Okonechnikov <`k.okonechnikov@gmail.com`>

Contents

aggregateBins 2
atacInferCnv 2
extractMetacells 3
plotCnvBlocks 4
prepareAtacInferCnvInput 4
runAtacInferCnv 6
writeConfig 7

Index **8**

`aggregateBins` *Combine ATAC signals across bins of specific size*

Description

Combine ATAC signals across bins of specific size

Usage

```
aggregateBins(sample, resDir, sId, bin_size, chrom_lengths)
```

Arguments

`sample`	Input Seurat object
`resDir`	Path to the result directory
`sId`	Result name
`bin_size`	Size of the bin e.g. 100000 for 100Kbp
`chrom_lengths`	Numeric vector of chromosome sizes, specific for genome

Value

Seurat object

`atacInferCnv` *atacInferCnv: CNV inference from single-cell chromatin accessibility data*

Description

`atacInferCnv` provides tools for genome-wide inference, analysis, and visualization of copy number variation (CNV) from single cell and single nucleus ATAC-seq data. The package extends the use of `infercnv` for chromatin accessibility profiles and includes utilities for preparing input matrices, aggregating genomic regions into bins, constructing meta-cells, running CNV inference, and visualizing inferred CNV profiles.

Details

The package is intended for the analysis of tumor single-cell chromatin accessibility datasets and supports identification of large-scale chromosomal alterations, and subclonal CNV structure.

Value

The package-level help page is used for documentation only and does not return a value.

Key functions

`prepareAtacInferCnvInput` Prepare input matrices and annotations for CNV inference from scATAC-seq data.

`runAtacInferCnv` Run CNV calling.

`plotCnvBlocks` Visualize inferred CNV block profiles.

`extractMetacells` *Compute meta-cells out of the main*

Description

Compute meta-cells out of the main

Usage

```
extractMetacells(
  resDir,
  sId,
  sample,
  targColumn,
  metacell_content = 5,
  verbose = TRUE
)
```

Arguments

`resDir`	Path to the result directory
`sId`	Result name
`sample`	Input Seurat object to split
`targColumn`	Name of annotation column to split
`metacell_content`	Amount of cells for adjustment, default n=5
`verbose`	Detailed output, progress messages, default TRUE

Value

Invisibly returns NULL.

`plotCnvBlocks` *Function to plot CNV blocks*

Description

This function creates a plot for CNV assigned/identified subclones

Usage

```
plotCnvBlocks(resDir, infercnvObj = NULL, save = FALSE, verbose = TRUE)
```

Arguments

`resDir`	Path to the result directory with input
`infercnvObj`	InferCnv result object. Default: NULL If NULL, then the object will be loaded from result directory.
`save`	Set TRUE to save plot to the result directory. Default: FALSE
`verbose`	Detailed output, progress messages, default TRUE

Value

Returns a list of plots

Examples

```
resPath = tempfile()
inPath = system.file("extdata", "MB183_ATAC_subset.tsv.gz",
                      package = "atacInferCnv")
sAnn = system.file("extdata", "MB183_ATAC_subset.CNV_blocks_ann_n30.txt",
                      package = "atacInferCnv")
prepareAtacInferCnvInput(inPath,sAnn,resPath, targColumn = "cnvBlock",
                        ctrlGrp = "Normal", performGA = FALSE)
runAtacInferCnv(resPath)
plotCnvBlocks(resPath)
```

`prepareAtacInferCnvInput` *Prepare input for the CNV calling from scATAC-seq data*

Description

Prepare input for the CNV calling from scATAC-seq data

Usage

```
prepareAtacInferCnvInput(
  dataPath = "",
  annPath = "",
  resDir = "",
  inObj = NULL,
  sId = "sample",
  targColumn = "CellType",
  ctrlGrp = "Normal",
  ctrlObj = NULL,
  binSize = NULL,
  chromLength = NULL,
  metaCells = FALSE,
  performGA = TRUE,
  verbose = TRUE
)
```

Arguments

`dataPath`	Path to the input data in 10X format or ATAC counts matrix in txt format (gzipped)
`annPath`	Path to annotation of the cells in tab-delimited format
`resDir`	Path to the result directory
`inObj`	Pre-computed Seurat/Signac or SingleCellExperiment object with required input data within (alternative for dataPath)
`sId`	Result name. Default: "Sample"
`targColumn`	Name of the target column in annotation. Default:CellType
`ctrlGrp`	Name for the reference control cell type. Could be several names, separated by comma. Default: "Normal"
`ctrlObj`	Seurat/Signac or SingleCellExperiment object to use as non-tumor control. Default:NULL
`binSize`	Apply custom bin size to combine signals in windows for CNV calling e.g. 500000 for 500 KBp. Default: NULL (not use this option)
`chromLength`	Numeric vector of chromosome sizes, specific for genome. Default: NULL
`metaCells`	Set TRUE to use meta cells (n=5 cells by default) or assign a number of cells. Default: FALSE
`performGA`	Perform general analysis of scATAC-data (clustering, UMAP). Default: TRUE
`verbose`	Detailed output, progress messages, and diagnostic information. Default: TRUE

Value

Invisibly returns NULL.

Examples

```
resPath = tempfile()
inPath = system.file("extdata", "MB183_ATAC_subset.tsv.gz",
                    package = "atacInferCnv")
sAnn = system.file("extdata", "MB183_ATAC_subset.CNV_blocks_ann_n30.txt",
```

```
                   package = "atacInferCnv" )
prepareAtacInferCnvInput(inPath,sAnn,resPath, targColumn = "cnvBlock",
                       ctrlGrp = "Normal",performGA = FALSE)
```

`runAtacInferCnv` *Wrapper function to run InferCNV calling*

Description

This function calls InferCNV from generated input. It has support for all original inferCnv options.

Usage

```
runAtacInferCnv(
  resDir,
  configFile = "infercnv_config.yml",
  numClusters = 1,
  chrToExclude = c("Y", "MT"),
  addDenoise = TRUE,
  clusterRefs = FALSE,
  smoothMethod = "runmeans",
  verbose = TRUE,
  returnObj = FALSE,
  ...
)
```

Arguments

`resDir`	Path to the result directory with input
`configFile`	Name of configuration file with InferCnv input data
`numClusters`	Number of clusters for hierarchical clustering. If equals one (by default) then no clustering is performed and provided annotation used for the formation of CNV groups.
`chrToExclude`	Chromosomes to exclude. Default: Y,MT
`addDenoise`	Activate denoise (InferCNV param). Default: TRUE
`clusterRefs`	Cluster also reference (InferCNV param). Default: FALSE
`smoothMethod`	Method for smoothing (InferCNV param). Default: runmeans
`verbose`	Detailed output, progress message and diagnostic information. If deactivated InferCnv log is saved to file infercnv.log. Default: TRUE
`returnObj`	Return InferCNV object to work with further. Default: FALSE
`...`	Other parameters to provide for infercnv::run, more details in documentation of this function

Value

Invisibly returns NULL by default.

Examples

```
resPath = tempfile()
inPath = system.file("extdata", "MB183_ATAC_subset.tsv.gz",
                      package = "atacInferCnv")
sAnn = system.file("extdata", "MB183_ATAC_subset.CNV_blocks_ann_n30.txt",
                      package = "atacInferCnv" )
prepareAtacInferCnvInput(inPath,sAnn,resPath, targColumn = "cnvBlock",
                         ctrlGrp = "Normal",performGA = FALSE)
resObj <- runAtacInferCnv(resPath, returnObj = TRUE)
```

`writeConfig` *Write InferCNV configuration*

Description

Write InferCNV configuration

Usage

```
writeConfig(resDir, sId, ctrlGrp = "Normal", binSize = NULL, meta = FALSE)
```

Arguments

`resDir`	Result directory path
`sId`	Result name
`ctrlGrp`	Name for the reference control cell type
`binSize`	Size of the bin e.g. 100000 for 100 Kbp
`meta`	True if use meta cells, default false

Value

Invisibly returns NULL.

Index

∗ **package**
    `atacInferCnv`, 2

`aggregateBins`, 2
`atacInferCnv`, 2
`atacInferCnv-package` *(*`atacInferCnv`*)*, 2

`extractMetacells`, 3

`plotCnvBlocks`, *3*, 4
`prepareAtacInferCnvInput`, *3*, 4

`runAtacInferCnv`, *3*, 6

`writeConfig`, 7