Package 'cinaR'

January 5, 2021

Type Package

Title A Computational Pipeline for Bulk 'ATAC-Seq' Profiles

Version 0.2.0

Description Differential analyses and Enrichment pipeline for bulk 'ATAC-seq' data analyses. This package combines different packages to have an ultimate package for both data analyses and visualization of 'ATAC-seq' data.

License GPL-3

Encoding UTF-8

LazyData true

URL https://github.com/eonurk/cinaR/

BugReports https://github.com/eonurk/cinaR/issues/

biocViews

Depends R (>= 3.5.0)

Imports ChIPseeker, DESeq2, dplyr, edgeR, fgsea, GenomicRanges, ggplot2, ggrepel, grDevices, limma, utils, pheatmap, preprocessCore, RColorBrewer, sva, TxDb.Hsapiens.UCSC.hg38.knownGene, TxDb.Hsapiens.UCSC.hg19.knownGene, TxDb.Mmusculus.UCSC.mm10.knownGene, writexl

RoxygenNote 7.1.1

Suggests biomaRt, knitr, rmarkdown

VignetteBuilder knitr

NeedsCompilation no

Author Onur Karakaslar [aut, cre], Duygu Ucar [aut] (<https://orcid.org/0000-0002-9772-3066>)

Maintainer Onur Karakaslar <eonurkara@gmail.com>

Repository CRAN

Date/Publication 2021-01-05 21:30:02 UTC

R topics documented:

`annotatePeaks` *annotatePeaks*

Description

Runs DA pipeline and makes it ready for enrichment analyses

Usage

```
annotatePeaks(cp, reference.genome, show.annotation.pie = FALSE)
```

Arguments

`cp`	bed formatted consensus peak matrix: CHR, START, STOP and raw peak counts (peaks by 3+samples)
`reference.genome`	genome of interested species. It should be 'hg38', 'hg19' or 'mm10'.
`show.annotation.pie`	shows the annotation pie chart produced with ChipSeeker

Value

DApeaks returns DA peaks

bed *Example peaks from bone marrow of B6 mice*

Description

Example peaks from bone marrow of B6 mice

Usage

```
data(atac_seq_consensus_bm)
```

Format

An object of class `data.frame` with 1000 rows and 25 columns.

Examples

```
data(atac_seq_consensus_bm)
```

cinaR *cinaR*

Description

Runs differential analyses and enrichment pipelines

Usage

```
cinaR(
  matrix,
  contrasts,
  experiment.type = "ATAC-Seq",
  DA.choice = 1,
  DA.fdr.threshold = 0.05,
  DA.lfc.threshold = 0,
  save.DA.peaks = FALSE,
  DA.peaks.path = NULL,
  norm.method = "cpm",
  filter.method = "custom",
  library.threshold = 2,
  cpm.threshold = 1,
  TSS.threshold = 50000,
  show.annotation.pie = FALSE,
  reference.genome = NULL,
  batch.correction = FALSE,
  batch.information = NULL,
```

```
  additional.covariates = NULL,
  run.enrichment = TRUE,
  enrichment.method = NULL,
  enrichment.FDR.cutoff = 1,
  background.genes.size = 20000,
  geneset = NULL
)
```

Arguments

matrix	either bed formatted consensus peak matrix (peaks by 3+samples) CHR, START, STOP and raw peak counts OR count matrix (genes by 1+samples).
contrasts	user-defined contrasts for comparing samples
experiment.type	The type of experiment either set to "ATAC-Seq" or "RNA-Seq"
DA.choice	determines which pipeline to run: (1) edgeR, (2) limma-voom, (3) limma-trend, (4) DEseq2. Note: Use limma-trend if consensus peaks are already normalized, otherwise use other methods.
DA.fdr.threshold	fdr cut-off for differential analyses
DA.lfc.threshold	log-fold change cutoff for differential analyses
save.DA.peaks	saves differentially accessible peaks to an excel file
DA.peaks.path	the path which the excel file of the DA peaks will be saved, if not set it will be saved to current directory.
norm.method	normalization method for consensus peaks
filter.method	filtering method for low expressed peaks
library.threshold	number of libraries a peak occurs so that it is not filtered default set to 2
cpm.threshold	count per million threshold for not to filter a peak
TSS.threshold	Distance to transcription start site in base-pairs. Default set to 50,000.
show.annotation.pie	shows the annotation pie chart produced with ChipSeeker
reference.genome	genome of interested species. It should be 'hg38', 'hg19' or 'mm10'.
batch.correction	logical, if set will run unsupervised batch correction via sva (default) or if the batch information is known 'batch.information' argument should be provided by user.
batch.information	character vector, given by user.
additional.covariates	vector or data.frame, this parameter will be directly added to design matrix before running the differential analyses, therefore won't affect the batch corrections but adjust the results in down-stream analyses.

`run.enrichment` logical, turns off enrichment pipeline

`enrichment.method`
There are two methodologies for enrichment analyses, Hyper-geometric p-value (HPEA) or Geneset Enrichment Analyses (GSEA).

`enrichment.FDR.cutoff`
FDR cut-off for enriched terms, p-values are corrected by Benjamini-Hochberg procedure

`background.genes.size`
number of background genes for hyper-geometric p-value calculations. Default is 20,000.

`geneset` Pathways to be used in enrichment analyses. If not set vp2008 (Chaussabel, 2008) immune modules will be used. This can be set to any geneset using 'read.gmt' function from 'qusage' package. Different modules are available: https://www.gsea-msigdb.org/gsea/downloads.jsp.

Value

returns differentially accessible peaks

Examples

```
data(atac_seq_consensus_bm) # calls 'bed'

# a vector for comparing the examples
contrasts <- sapply(strsplit(colnames(bed), split = "-", fixed = TRUE),
                    function(x){x[1]})[4:25]

results <- cinaR(bed, contrasts, reference.genome = "mm10")
```

`color_values` *color values*

Description

color values

Usage

```
color_values
```

Format

An object of class `character` of length 8.

`differentialAnalyses` *Differential Analyses*

Description

Runs differential analyses pipeline of choice on consensus peaks

Usage

```
differentialAnalyses(
  final.matrix,
  contrasts,
  experiment.type,
  DA.choice,
  DA.fdr.threshold,
  DA.lfc.threshold,
  save.DA.peaks,
  DA.peaks.path,
  batch.correction,
  batch.information,
  additional.covariates
)
```

Arguments

`final.matrix`	Annotated Consensus peaks
`contrasts`	user-defined contrasts for comparing samples
`experiment.type`	The type of experiment either set to "ATAC-Seq" or "RNA-Seq"
`DA.choice`	determines which pipeline to run: (1) edgeR, (2) limma-voom, (3) limma-trend, (4) DEseq2
`DA.fdr.threshold`	fdr cut-off for differential analyses
`DA.lfc.threshold`	log-fold change cutoff for differential analyses
`save.DA.peaks`	logical, saves differentially accessible peaks to an excel file
`DA.peaks.path`	the path which the excel file of the DA peaks will be saved, if not set it will be saved to current directory.
`batch.correction`	logical, if set will run unsupervised batch correction via sva (default) or if the batch information is known ‘batch.information‘ argument should be provided by user.
`batch.information`	character vector, given by user.

additional.covariates
: vector or data.frame, this parameter will be directly added to design matrix before running the differential analyses, therefore won't affect the batch corrections but adjust the results in down-stream analyses.

Value

returns consensus peaks (batch corrected version if enabled) and DA peaks

---

dot_plot *dot_plot*

Description

Given the results from 'cinaR' it produces dot plots for enrichment analyses.

Usage

```
dot_plot(results, fdr.cutoff = 0.1, filter.pathways = FALSE)
```

Arguments

results	cinaR result object
fdr.cutoff	Pathways with smaller fdr values than the cut-off will be shown as dots.
filter.pathways	logical, it will filter the pathways from dot plot with fdr values less than 'fdr.cutoff'.

Value

ggplot object

Examples

```
library(cinaR)
data(atac_seq_consensus_bm) # calls 'bed'

# a vector for comparing the examples
contrasts <- sapply(strsplit(colnames(bed), split = "-", fixed = TRUE),
                    function(x){x[1]})[4:25]

results <- cinaR(bed, contrasts, reference.genome = "mm10")

dot_plot(results)
```

filterConsensus *filterConsensus*

Description

Filters lowly expressed peaks from down-stream analyses

Usage

```
filterConsensus(
  cp,
  filter.method = "custom",
  library.threshold = 2,
  cpm.threshold = 1
)
```

Arguments

cp	consensus peak matrix, with unique ids at rownames.
filter.method	filtering method for low expressed peaks
library.threshold	number of libraries a peak occurs so that it is not filtered default set to 2
cpm.threshold	count per million threshold for not to filter a peak

Value

returns differentially accessible peaks

Examples

```
set.seed(123)
cp <- matrix(rexp(200, rate=.1), ncol=20)

## using cpm function from `edgeR` package
cp.filtered <- filterConsensus(cp)
```

grch37 *Grch37*

Description

Grch37

Usage

```
data(grch37)
```

Format

An object of class `tbl_df` (inherits from `tbl`, `data.frame`) with 66978 rows and 3 columns.

---

grch38 *Grch38*

Description

Grch38

Usage

```
data(grch38)
```

Format

An object of class `tbl_df` (inherits from `tbl`, `data.frame`) with 67495 rows and 3 columns.

---

grcm38 *Grcm38*

Description

Grcm38

Usage

```
data(grcm38)
```

Format

An object of class `data.frame` with 25350 rows and 4 columns.

GSEA *GSEA Having run the differential analyses this function runs geneset enrihment analyses with 'fgsea' package.*

Description

GSEA Having run the differential analyses this function runs geneset enrihment analyses with 'fgsea' package.

Usage

```
GSEA(genes, geneset)
```

Arguments

genes	DA gene names to be checked if they are over-represented or not.
geneset	Pathways to be used in enrichment analyses. If not set vp2008 (Chaussabel, 2008) immune modules will be used. This can be set to any geneset using 'read.gmt' function from 'qusage' package. Different modules are available: https://www.gsea-msigdb.org/gsea/downloads.jsp.

References

G. Korotkevich, V. Sukhov, A. Sergushichev. Fast gene set enrichment analysis. bioRxiv (2019), doi:10.1101/060012

Examples

```
library(cinaR)
library(fgsea)
data(examplePathways)
data(exampleRanks)
GSEA(exampleRanks, examplePathways)
```

heatmap_differential *heatmap_differential*

Description

plot differentially accessible peaks for a given comparison

Usage

```
heatmap_differential(results, comparison = NULL, ...)
```

Arguments

results	cinaR result object
comparison	these are created by cinaR from 'contrasts' user provided. If not selected the first comparison will be shown!
...	additional arguments for heatmap function, for more info '?pheatmap'

Value

ggplot object

Examples

```
library(cinaR)
data(atac_seq_consensus_bm) # calls 'bed'

# a vector for comparing the examples
contrasts <- sapply(strsplit(colnames(bed), split = "-", fixed = TRUE),
                    function(x){x[1]})[4:25]

results <- cinaR(bed, contrasts, reference.genome = "mm10")

heatmap_differential(results)
```

---

heatmap_var_peaks *heatmap_var_peaks*

Description

plot most variable k peaks (default k = 100) among all samples

Usage

```
heatmap_var_peaks(results, heatmap.peak.count = 100, ...)
```

Arguments

results	cinaR result object
heatmap.peak.count	number of peaks to be plotted. If number of peaks are less than k then all peaks will be used.
...	additional arguments for heatmap function, for more info '?pheatmap'

Value

ggplot object

Examples

```
library(cinaR)
data(atac_seq_consensus_bm) # calls 'bed'

# creating dummy results
results <- NULL
results[["cp"]] <- bed[,c(4:25)]

heatmap_var_peaks(results)
```

HPEA *HPEA*

Description

HPEA

Usage

```
HPEA(genes, geneset, background.genes.size)
```

Arguments

genes	DA gene names to be checked if they are over-represented or not.
geneset	Pathways to be used in enrichment analyses. If not set vp2008 (Chaussabel, 2008) immune modules will be used. This can be set to any geneset using 'read.gmt' function from 'qusage' package. Different modules are available: https://www.gsea-msigdb.org/gsea/downloads.jsp.
background.genes.size	number of background genes for hyper-geometric p-value calculations. Default is 20,000.

Examples

```
library(cinaR)

data("VP2008")
genes.to.test <- vp2008[[1]][1:10]
HPEA(genes.to.test,vp2008, background.genes.size = 20e3)
```

mouse2human *mouse2human*

Description

Given the mice gene symbols, this function creates a map from mice to human using biomaRt.

Usage

```
mouse2human(genes)
```

Arguments

genes	mice genes to be converted to human

Value

returns a mapping from mouse to human

Examples

```
mouse.genes <- c("Gfap", "Gzmb", "Il1b")
map <- mouse2human(mouse.genes)
```

normalizeConsensus *normalizeConsensus*

Description

Normalizes consensus peak using different methods

Usage

```
normalizeConsensus(cp, norm.method = "cpm", log.option = FALSE)
```

Arguments

cp	bed formatted consensus peak matrix: CHR, START, STOP and raw peak counts (peaks by 3+samples)
norm.method	normalization method for consensus peaks
log.option	logical, log option for cpm function in edgeR

Value

Normalized consensus peaks

Examples

```
set.seed(123)
cp <- matrix(rexp(200, rate=.1), ncol=20)

## using cpm function from `edgeR` package
cp.normalized <- normalizeConsensus(cp)

## quantile normalization option
cp.normalized <- normalizeConsensus(cp, norm.method = "quantile")
```

`pca_plot` *pca_plot*

Description

pca_plot

Usage

```
pca_plot(results, overlaid.info, sample.names = NULL, show.names = TRUE)
```

Arguments

`results`	cinaR result object
`overlaid.info`	overlaid information onto the samples
`sample.names`	names of the samples shown on pca plot
`show.names`	logical, if set FALSE sample names will be hidden

Value

ggplot object

Examples

```
#' library(cinaR)
data(atac_seq_consensus_bm) # calls 'bed'

# creating dummy results
results <- NULL
results[["cp"]] <- bed[,c(4:25)]

# a vector for comparing the examples
```

```
contrasts <- sapply(strsplit(colnames(bed), split = "-", fixed = TRUE),
                    function(x){x[1]})[4:25]

## overlays the contrasts info onto PCA plots
pca_plot(results, contrasts)

## you can overlay other information as well,
## as long as it is the same length with the
## number of samples.

sample.info <- c(rep("Group A", 11), rep("Group B", 11))
pca_plot(results, sample.info, show.names = FALSE)
```

`run_enrichment` *run_enrichment*

Description

This function is run, if the enrichment pipeline wants to be called afterwards. Setting reference genome to the same genome which cinaR was run should be given to this function!

Usage

```
run_enrichment(
  results,
  geneset = NULL,
  experiment.type = "ATAC-Seq",
  reference.genome = NULL,
  enrichment.method = NULL,
  enrichment.FDR.cutoff = 1,
  background.genes.size = 20000
)
```

Arguments

`results`	list, DA peaks list for different contrasts
`geneset`	Pathways to be used in enrichment analyses. If not set vp2008 (Chaussabel, 2008) immune modules will be used. This can be set to any geneset using ‘read.gmt‘ function from ‘qusage‘ package. Different modules are available: https://www.gsea-msigdb.org/gsea/downloads.jsp.
`experiment.type`	The type of experiment either set to "ATAC-Seq" or "RNA-Seq"
`reference.genome`	genome of interested species. It should be ’hg38’, ’hg19’ or ’mm10’.
`enrichment.method`	There are two methodologies for enrichment analyses, Hyper-geometric p-value (HPEA) or Geneset Enrichment Analyses (GSEA).

enrichment.FDR.cutoff
: FDR cut-off for enriched terms, p-values are corrected by Benjamini-Hochberg procedure

background.genes.size
: number of background genes for hyper-geometric p-value calculations. Default is 20,000.

Examples

```
library(cinaR)
data(atac_seq_consensus_bm) # calls 'bed'

# a vector for comparing the examples
contrasts <- sapply(strsplit(colnames(bed), split = "-", fixed = TRUE),
                    function(x){x[1]})[4:25]

results <- cinaR(bed, contrasts, reference.genome = "mm10", run.enrichment = FALSE)

results_with_enrichment <- run_enrichment(results, reference.genome = "mm10")
```

---

scale_rows *scale_rows*

Description

Normalize (z-score) rows of a matrix

Usage

```
scale_rows(x)
```

Arguments

x	a matrix, possibly containing gene by samples

Value

Row-normalized matrix

Examples

```
library(cinaR)
data(atac_seq_consensus_bm) # calls 'bed'
bed.row.normalized <- scale_rows(bed[,c(4:25)])
head(bed.row.normalized)
```

show_comparisons *show_comparisons*

Description

returns the names of the created comparisons

Usage

```
show_comparisons(results)
```

Arguments

results	output of the cinaR

Value

comparisons created

vp2008 *Immune modules*

Description

Immune modules

Usage

```
data(VP2008)
```

Format

An object of class GMT; see `read.gmt` from qusage package.

References

Chaussabel et al. (2008) Immunity 29:150-164 (PubMed)

Index