Package 'longitudinalDynamics'

October 14, 2021

Type Package

Title Identify Inter-, Intra-donor variations in bulk or single cell longitudinal dataset

Version 0.1.0

Author Suhas Vasaikar [aut, cre],
Aarthi Talla [aut, ctb],
Xiaojun Li [aut, ctb]

Maintainer Suhas Vasaikar <suhas.vasaikar@alleninstitute.org>

Description LongitudinalDynamics (longitudinalDynamics) is a platform for anayzing longitudinal data from bulk as well as single cell. It allows to identify inter-, intra-donor variations in genes over longitudinal time points. The analysis can be done on bulk expression dataset without known celltype information or single cell with celltype/user-defined groups. It allows to infer stable and variable features in given donor and each celltype (or user defined group). The outlier analysis can be performed to identify techinical/biological perturbed samples in donor/participant. Further, differential analysis can be performed to deciher time-wise changes in gene expression in a celtype.

Depends R (>= 3.5.0), methods, grid, graphics, stats, grDevices, ggplot2, reshape2, ComplexHeatmap, circlize, cowplot, pheatmap, tidyverse

Imports Seurat (>= 3.9),
ggrepel (>= 0.9),
pbapply (>= 1.4),
lme4 (>= 1.1),
ggforce (>= 0.3),
MAST (>= 1.14),
factoextra (>= 1.0),
Rtsne (>= 0.15),
knitr(>= 1.30),
dplyr

Suggests ArchR (>= 1.0),
rmarkdown

biocViews Data analysis, Longitudinal data, Single cell, scRNA, scATAC, Software, Visualization

License MIT + file LICENSE

Encoding UTF-8

LazyData true

Roxygen list(markdown = TRUE)

RoxygenNote 7.1.1

VignetteBuilder knitr

R topics documented:

`avgExpCalc` *A avgExpCalc Function*

Description

This function allows you to calculate average gene expression on long-normalized data by group defined by user

Usage

```
avgExpCalc(dataObj, assay = "RNA", group.by)
```

Arguments

`dataObj`	scRNA object with log-normalized data
`assay`	Single cell data Assay type ("RNA", "SCT"). Default "RNA"
`group.by`	Calculate average expression by given group

Examples

```
##Input Expression data
#avgExpCalc(dataObj, group.by)
```

cvCalcBulk *A cvCalcBulk Function*

Description

This function allows to calculate Intra-donor variations over longitudinal timepoints. The coefficient of variation (CV) is calculated in Bulk data without group information. CV calculated across samples. It requires longitudinal data matrix/data frame and annotation file.

Usage

```
cvCalcBulk(
  ann,
  mat,
  meanThreshold = NULL,
  cvThreshold,
  housekeeping_genes = NULL,
  fileName = NULL,
  filePATH = NULL
)
```

Arguments

ann	Annotation table. Table must consist column Sample (Participant sample name), PTID (Participant), Time (longitudinal time points)
mat	Expression matrix or data frame. Rows represents gene/proteins column represents participant samples (same as annotation table Sample column)
meanThreshold	Average expression threshold to filter lowly expressed genes Default is 0.1 (log2 scale)
cvThreshold	Coefficient of variation threshold to select variable and stable genes Default is 5 for bulk olink data
housekeeping_genes	Optional list of housekeeping genes to focus on. Default is ACTB, GAPDH
fileName	User-defined file name, Default outputFile
filePATH	User-defined output directory PATH Default, current directory

cvCalcBulkProfile *A cvCalcBulkProfile Function*

Description

This function allows to calculate Intra-donor variations over longitudinal timepoints. The coefficient of variation (CV) is calculated in Bulk data without group information. CV calculated across samples. It requires longitudinal data matrix/data frame and annotation file.

Usage

```
cvCalcBulkProfile(ann, mat, fileName = NULL, filePATH = NULL)
```

Arguments

ann	Annotation table. Table must consist column Sample (Participant sample name), PTID (Participant), Time (longitudinal time points)
mat	Expression matrix or data frame. Rows represents gene/proteins column represents participant samples (same as annotation table Sample column)
fileName	User-defined file name, Default outputFile
filePATH	User-defined output directory PATH Default, current directory

---

cvCalcSC *A cvCalcSC Function*

Description

This function allows to calculate Intra-donor variations over longitudinal timepoints. The coefficient of variation is calculated in single cell data. It requires longitudinal data matrix/data frame and annotation file.

Usage

```
cvCalcSC(
  ann,
  mat,
  meanThreshold = NULL,
  cvThreshold = NULL,
  housekeeping_genes = NULL,
  fileName = NULL,
  filePATH = NULL
)
```

Arguments

ann	Annotation table. Table must consist column Sample (Participant sample name), PTID (Participant), Time (longitudinal time points)
mat	Expression matrix or data frame. Rows represents gene/proteins column represents participant samples (same as annotation table Sample column)
meanThreshold	Average expression threshold to filter lowly expressed genes Default is 0.1 (log2 scale)
cvThreshold	Coefficient of variation threshold to select variable and stable genes Default is 10 for single cell RNA (100*SD/mean)
housekeeping_genes	Optional list of housekeeping genes to focus on. Default is ACTB, GAPDH
fileName	User-defined file name, Default outputFile
filePATH	User-defined output directory PATH Default, current directory

`cvprofile` *A cvprofile Function*

Description

This function allows to calculate Intra-donor variations over longitudinal timepoints. The coefficient of variation is calculated in single cell data. It requires longitudinal data matrix/data frame and annotation file.

Usage

```
cvprofile(
  mat,
  ann,
  meanThreshold = NULL,
  housekeeping_genes = NULL,
  fileName = NULL,
  filePATH = NULL
)
```

Arguments

`mat`	Expression matrix or data frame. Rows represents gene/proteins column represents participant samples (same as annotation table Sample column)
`ann`	Annotation table. Table must consist column Sample (Participant sample name), PTID (Participant), Time (longitudinal time points)
`meanThreshold`	Average expression threshold to filter lowly expressed genes Default is 0.1 (log2 scale)
`housekeeping_genes`	Optional list of housekeeping genes to focus on. Default is ACTB, GAPDH
`fileName`	User-defined file name, Default outputFile
`filePATH`	User-defined output directory PATH Default, current directory

`cvSampleprofile` *A cvSampleprofile Function*

Description

This function allows to calculate Intra-donor variations over longitudinal timepoints. The coefficient of variation is calculated in single cell data. It requires longitudinal data matrix/data frame and annotation file.

Usage

```
cvSampleprofile(
  mat,
  ann,
  meanThreshold = NULL,
  cvThreshold = NULL,
  fileName = NULL,
  filePATH = NULL
)
```

Arguments

mat	Expression matrix or data frame. Rows represents gene/proteins column represents participant samples (same as annotation table Sample column)
ann	Annotation table. Table must consist column Sample (Participant sample name), PTID (Participant), Time (longitudinal time points)
meanThreshold	Average expression threshold to filter lowly expressed genes Default is 0.1 (log2 scale)
cvThreshold	Coefficient of variation threshold to select variable and stable genes Default is 10 for single cell RNA (100*SD/mean)
fileName	User-defined file name, Default outputFile
filePATH	User-defined output directory PATH Default, current directory

---

dimUMAPPlot *A dimUMAPPlot Function*

Description

This function allows you to perform UMAP visualization of gene of interest list.

Usage

```
dimUMAPPlot(
  ann,
  rnaObj = NULL,
  countMat = NULL,
  nPC = 30,
  gene_oi = NULL,
  groupName = NULL,
  plotname = NULL,
  filePATH = NULL,
  fileName = NULL
)
```

Arguments

ann	Annotation table. Table must consist column Sample (Participant sample name), PTID (Participant), Time (longitudinal time points), group, name of the group, group_donor (combined string using group:Sample)

rnaObj	The seurat scRNA object in case of single cell RNA data (optional).
countMat	Expression matrix or data frame. Rows represents gene/proteins column represents participant samples (same as annotation table Sample column), in case count matrix for expression data (optional).
nPC	Number of PCAs to be used for UMAP, Default is 30
gene_oi	Genes of interest to explore, required
groupName	User-defined group name column from annotation table or seurat annotation column, required
plotname	User-defined output file name, required
filePATH	User-defined output directory PATH Default, current directory
fileName	User-defined file name, Default outputFile

Examples

```
##Count/genescore matrix data
#dimUMAPPlot(ann=annotation, countMat=countData, nPC=15, gene_oi=var_gene,
#groupName="celltype", plotname="variable", filePATH=filePATH, fileName="ATAC")

##Single cell RNA data
#dimUMAPPlot(rnaObj=SeuratObj, nPC=15, gene_oi=var_gene, groupName="celltype",
#plotname="variable", filePATH=filePATH, fileName="scRNA")
```

---

genecircosPlot *A genecircosPlot Function*

Description

This function allows you to Circos Plot for gene list of interest by group

Usage

```
genecircosPlot(
  data,
  geneList,
  groupColumn = NULL,
  groupBy = NULL,
  colorThreshold = NULL
)
```

Arguments

data	Expression matrix or data frame. Rows represents gene/proteins column represents group:donor (group and donor separated by :)
geneList	Genes of interest to explore
groupColumn	Default 1, use 2 when columns are donor:group format
groupBy	Optional, User-defined groups to consider and order
colorThreshold	User-defined color threshold in colorspace

Examples

```
##Circos Plot for genes expression in a group
#geneList <- c("IL32","CCL5","TCF7","IL7R","LEF1")
#res <- genecircosPlot(data=cv_res, geneList=geneList)
```

---

genePlot *A genePlot Function*

Description

This function allows you to perform UMAP visualization of gene of interest list.

Usage

```
genePlot(ann, data, geneName, groupName = NULL)
```

Arguments

ann	Annotation table. Table must consist column Sample (Participant sample name), PTID (Participant), Time (longitudinal time points), group, name of the group, group_donor (combined string using group:Sample)
data	Average Expression matrix or data frame. Rows represents gene/proteins column represents participant samples with group (optional).
geneName	User-defined gene name
groupName	User-defined group name column from annotation table

Examples

```
#plot <- genePlot(ann=annotation, data=ExpressionData, geneName="FOLR3", groupName="Time")
```

---

lmeVariance *A lmeVariance Function*

Description

This function allows you to calculate inter-donor variation between participants over longitudinal time points. It uses linear mixed model to calculate variance contribution from each given feature list

Usage

```
lmeVariance(
  ann,
  mat,
  featureSet,
  meanThreshold = NULL,
  fileName = NULL,
  filePATH = NULL
)
```

Arguments

ann	Annotation table. Table must consist column Sample (Participant sample name), PTID (Participant), Time (longitudinal time points)
mat	Expression matrix or data frame. Rows represents gene/proteins column represents participant samples (same as annotation table Sample column)
featureSet	Variance analysis carried out for the feature set provided such as c("PTID", "Time", "Sex")
meanThreshold	Average expression threshold to filter lowly expressed genes/features Default is 0
fileName	User-defined file name, Default outputFile
filePATH	User-defined output directory PATH Default, current directory

Examples

```
##Input Expression data
#filePATH <- getwd()
#lmem_res <- lmeVariance(ann=metadata, mat=datamatrix,
#featureSet=c("PTID", "Time", "Sex"),
#meanThreshold=0.1, fileName="RNA", filePATH=filePATH)
```

---

longitudinalDynamics *A longitudinalDynamics Function*

Description

This function allows you to perform analysis of longitudinal dataset. It requires longitudinal data matrix/data frame and annotation file.

Usage

```
longitudinalDynamics(
  metadata = NULL,
  data = NULL,
  datatype = NULL,
  omics = NULL,
  featureSet = NULL,
  meanThreshold = 1,
  cvThreshold = 5,
  NA_threshold = 0.4,
  column_sep = NULL,
  coding_genes = NULL,
  avgGroup = NULL,
  housekeeping_genes = c("ACTB", "GAPDH"),
  group_oi = NULL,
  nPC = 15,
  donorThreshold = NULL,
  groupThreshold = NULL,
  topFeatures = 25,
  method = "spearman",
```

```
  clusterBy = "donor",
  SD_threshold = 2,
  doOutlier = FALSE,
  fileName = NULL,
  outputDirectory = NULL
)
```

Arguments

metadata	Annotation table. Table must consist column Sample (Participant sample name), PTID (Participant), Time (longitudinal time points)
data	Expression matrix or data frame. Rows represents gene/proteins column represents participant samples (same as annotation table Sample column). For single cell, Single cell RNA Seurat object, if datatype is single cell RNA and Single cell ATAC genescore matrix or data frame
datatype	Data input can be bulk or singlecell
omics	User defined name like RNA, ATAC, Proteomics, FLOW
featureSet	Variance analysis carried out on the featureSet provided such as c("PTID", "Time", "Sex")
meanThreshold	Average expression threshold to filter lowly expressed genes Default is 0.1 (log2 scale)
cvThreshold	Coefficient of variation threshold to select variable and stable genes Default is 10 for single cell RNA (100*SD/mean)
NA_threshold	Number of NAs in data (numeric value or NULL). Default, 40% * number of columns.
column_sep	Separator of "PTID" and "Time" in "Sample" column of Annotation table like column_sep="W" for PTID1W1, column_sep=":" for PTID1W1:Tcell
coding_genes	Selecting protein coding/user-defined gene list only
avgGroup	Group label to be used to calculate average gene expression by group label
housekeeping_genes	Optional list of housekeeping genes to focus on Default is NULL
group_oi	Group of interest to focus on, Default is NULL
nPC	Number of PCAs to be used for UMAP, Default is 15
donorThreshold	Donor threshold number to be used, Default is number of participants
groupThreshold	Group label threshold number to be used, Default is (number of participants x group labels)/2
topFeatures	Number of features to be selected from each group, Default is 25
method	Sample correlation analysis ("pearson", "spearman"). Default is "spearman"
clusterBy	for sample correlation cluster columns by ("donor", "group")
SD_threshold	Standard deviation limit to find outliers (Eg. SD_threshold= 2, equals to Mean+/-2SD)
doOutlier	To perform outlier analysis (TRUE or FALSE). Default FALSE
fileName	User defined filename
outputDirectory	User-defined output directory Default, output

multimodalView *A multimodalView Function*

Description

This function allows you to vizualize the multimodal view genes of interest by celltypes/ groups defined by use

Usage

```
multimodalView(
  modality1,
  modality2,
  groupBy = NULL,
  geneList,
  colorThreshold = 10,
  groupColumn = NULL,
  plotHeight = 10,
  fileName = NULL,
  filePATH = NULL
)
```

Arguments

modality1	Variation or Expression matrix/data frame. Rows represents gene/proteins column represents group:donor (group and donor separated by :)
modality2	Variation or Expression matrix/data frame. Rows represents gene/proteins column represents group:donor (group and donor separated by :)
groupBy	Optional, User-defined groups to consider and order
geneList	Genes of interest to explore
colorThreshold	User-defined color threshold in colorspace
groupColumn	Default 1, use 2 when columns are donor:group format
plotHeight	User-defined Plot size (in)
fileName	User defined filename
filePATH	User-defined output directory path to save result

Examples

```
##Circos Plot for genes expression in a group
#geneList <- c("HLA-A","HLA-B","HLA-C","HLA-DRA","HLA-DPA1","HLA-DRB1")
#multimodalView(modality1=scrna_cv_res, modality2=scatac_cv_res, geneList)
```

`outlierDetect` *A outlierDetect Function*

Description

This function allows you to perform outlier analysis on bulk data by calculating z-score. Outlier genes defined as (exp-avgExp/SD) > mean + 2SD or (exp-avgExp/SD) < mean - 2SD.

Usage

```
outlierDetect(
  ann,
  mat,
  SD_threshold = 2,
  plotWidth = 10,
  plotHeight = 5,
  groupBy = FALSE,
  fileName = NULL,
  filePATH = NULL
)
```

Arguments

`ann`	Annotation table. Table must consist column Sample (Participant sample name), PTID (Participant), Time (longitudinal time points)
`mat`	Expression matrix or data frame. Rows represents gene/proteins column represents participant samples (same as annotation table Sample column)
`SD_threshold`	Standard deviation limit to find outlliers (Eg. SD_threshold= 2, equals to Mean+/-2SD)
`plotWidth`	User-defined plot width, Default 10 in
`plotHeight`	User-defined plot height, Default 5 in
`groupBy`	Include groupwise outlier analysis (TRUE or FALSE)
`fileName`	User-defined file name, Default outputFile
`filePATH`	User-defined output directory PATH Default, current directory

Examples

```
#filePATH <- getwd()
#outlier_res <- outlierDetect(ann=metadata, mat=datamatrix)
```

`sample_correlation` *A sample_correlation Function*

Description

This function allows to perform sample correlation (by group like celltype, ot by donor).

Usage

```
sample_correlation(
  data,
  column_sep = ":",
  method = "spearman",
  groupColumn = 2,
  clusterBy = "donor",
  max = 0.9,
  column_names_fontsize = 4,
  row_names_fontsize = 4,
  row_title_fontsize = 6,
  column_title_fontsize = 6,
  plotHeight = 20,
  fileName = NULL,
  filePATH = NULL
)
```

Arguments

`data`	Expression matrix or data frame. Rows represents gene/proteins column represents participant samples (if celltype with in donor then sample: celltype, separated by :)
`column_sep`	Sample and celltype seperator like (:)
`method`	Correlation method "pearson" or "spearman"
`groupColumn`	Data column names consists group (Donor-group) at 2nd place or 1st place(like PTIDxGroupX, 2 or GroupXPTIDx, 1)
`clusterBy`	Cluster correlation result by "donor" or "group". Default donor
`max`	Maximum color limit (Default, 0.9 correlation)
`column_names_fontsize`	Font size of the column names, default 4
`row_names_fontsize`	Font size of the row names, default 4
`row_title_fontsize`	Font size of the row title, default 6
`column_title_fontsize`	Font size of the column title, default 6
`plotHeight`	Height of the plot (in), deafult 20in
`fileName`	User-defined file name, Default outputFile
`filePATH`	User-defined output directory PATH Default, current directory

Examples

```
#res <- sample_correlation(data=datamatrix, column_sep=":", method="spearman")
```

---

scatac_archr_genescore *A scatac_archr_genescore Function*

Description

This function allows you to calculate genescore matrix from scATAC archR object. This function requires archR package installed and scATAC object created.

Usage

```
scatac_archr_genescore(ArchRProj, groupBy)
```

Arguments

ArchRProj	archR scATAC object for input single cell ATAC longitudinal data
groupBy	Group label to be used to calculate average gene expression by group label, Eg. "celltype"

Examples

```
##Input scATAC data
#genescore <- scatac_archr_genescore(ArchRProj=proj, groupBy="celltype")
```

---

sclongitudinalDGE *A sclongitudinalDGE Function*

Description

This function allows you to calculate differential expressed genes in the direction of given time points (if timepoints>3 otherwise DEGs between two timepoints). A hurdle model was fit to each participant independently in order to identify participant-specific longitudinal transcriptomic changes. Genes that were expressed in at least 10% of cells per participant were considered for this analysis. The models were fit on the input normalized data, modeling the timepoints as a continuous variable within each cell type and adjusting for the batch only if any timepoints from the same participant were run across multiple batches.

Usage

```
sclongitudinalDGE(
  ann,
  dataObj,
  scassay = "RNA",
  celltypecol,
  mincellsexpressed = 0.1,
  removelnc = "TRUE",
```

```
  adjfac = "none",
  baseline = NULL,
  plotWidth = 10,
  plotHeight = 10,
  fileName = NULL,
  filePATH = NULL
)
```

Arguments

ann	Annotation dataframe. Table must consist column Sample (Participant sample name), PTID (Participant), Time (longitudinal time points)
dataObj	Single cell RNA seurat object. Seurat object should have column name Sample (same as annotation table Sample column)
scassay	Single cell assay from scRNA seurat object (Default "RNA")
celltypecol	Column of interest such as celltype to analyze DEGs in participant over time
mincellsexpressed	Average expression threshold to filter lowly expressed genes/features Default is 0.1
removelnc	Remove lincRNAs, mitochondrial and ribosomal genes from analysis incldes (^RP|^MT-|^LINC|orf) (TRUE/FALSE). Default is TRUE
adjfac	Factors to be adjusted for such as batch, sex
baseline	Donors (PTID) to be considered as baseline. Deafult NULL
plotWidth	User-defined plot width, Default 10 in
plotHeight	User-defined plot height, Default 10 in
fileName	User-defined file name, Default outputFile
filePATH	User-defined output directory PATH Default, current directory

Examples

```
##Input scRNA data and annotation file
#DEGres <- sclongitudinalDGE(ann=metadata, dataObj=pbmc, scassay="RNA", celltypecol="celltype")
```

StableFeatures *A StableFeatures Function*

Description

This function allows you to identify stable genes in a participant across longitudinal timepoints in single cell dataset. The coefficient of variation (CV) obtained from 'cvCalcSC' function used to filter genes/features by CV threshold (cvThreshold). User can identify cvThreshold in different datasets using housekeeping genes CV distribution. The minimum expression of gene (meanThreshold) used to remove lowly expressed genes (spike CV).

Usage

```
StableFeatures(
  ann = NULL,
  group_oi = NULL,
  meanThreshold = NULL,
  cvThreshold = NULL,
  donorThreshold = NULL,
  groupThreshold = NULL,
  topFeatures = 25,
  filePATH = NULL,
  fileName = NULL
)
```

Arguments

ann	Annotation table. Table must consist column Sample (Participant sample name), PTID (Participant), Time (longitudinal time points)
group_oi	Group of interest to focus on, Default is NULL
meanThreshold	Average expression threshold to filter lowly expressed genes Default is 0.1 (log2 scale)
cvThreshold	Coefficient of variation threshold to select variable and stable genes Default is 10 for single cell RNA (100*SD/mean)
donorThreshold	Donor threshold number to be used, Default is number of participants
groupThreshold	Group label threshold number to be used, Default is (number of participants x group labels)/2
topFeatures	Number of features to be selected from each group, Default is 25
filePATH	User-defined output directory path to load the CV result obtained from cvCalcSC function
fileName	User defined filename

Examples

```
##Single cell RNA data
#stablegene <- StableFeatures(ann=metadata, meanThreshold=0.1, cvThreshold=10,
#donorThreshold=donorThreshold, groupThreshold=groupThreshold,
#topFeatures=25, fileName="scRNA", filePATH=filePATH)
```

VarFeatures *A VarFeatures Function*

Description

This function allows you to identify variable genes in a participant across longitudinal timepoints in single cell dataset. The coefficient of variation (CV) obtained from 'cvCalcSC' function used to filter genes/features by CV threshold (cvThreshold). User can identify cvThreshold in different datasets using housekeeping genes CV distribution. The minimum expression of gene (meanThreshold) used to remove lowly expressed genes (spike CV).

Usage

```
VarFeatures(
  ann = NULL,
  group_oi = NULL,
  meanThreshold = NULL,
  cvThreshold = NULL,
  donorThreshold = NULL,
  groupThreshold = NULL,
  topFeatures = 25,
  filePATH = NULL,
  fileName = NULL
)
```

Arguments

ann	Annotation table. Table must consist column Sample (Participant sample name), PTID (Participant), Time (longitudinal time points)
group_oi	Group of interest to focus on, Default is NULL
meanThreshold	Average expression threshold to filter lowly expressed genes Default is 0.1 (log2 scale)
cvThreshold	Coefficient of variation threshold to select variable and stable genes Default is 10 for single cell RNA (100*SD/mean)
donorThreshold	Donor threshold number to be used, Default is number of participants
groupThreshold	Group label threshold number to be used, Default is (number of participants x group labels)/2
topFeatures	Number of features to be selected from each group, Default is 25
filePATH	User-defined output directory path to load the CV result obtained from cvCalcSC function
fileName	User defined filename

Examples

```
#Single cell RNA data
#vargenes <- VarFeatures(ann=metadata, meanThreshold=0.1, cvThreshold=10,
#donorThreshold=donorThreshold, groupThreshold=groupThreshold,
#topFeatures=25, fileName="scRNA", filePATH="output/")
```