# Package: SingleCellStat (via r-universe)

May 8, 2026

**Type** Package

**Title** A Toolkit for Statistical Analysis of Single-Cell Omics Data

**Version** 0.3.1

**Date** 2025-04-28

**Description** A suite of statistical methods for analysis of single-cell omics data including linear model-based methods for differential abundance analysis for individual level single-cell RNA-seq data. For more details see Zhang, et al. (Submitted to Bioinformatics)<https://github.com/Lujun995/DiSC_Replication_Code>.

**Depends** R (>= 4.3.0)

**Suggests** knitr, rmarkdown

**Imports** matrixStats, Matrix, stats, utils, vegan

**License** GPL-3

**Encoding** UTF-8

**LazyData** true

**LazyDataCompression** xz

**NeedsCompilation** no

**Author** Lujun Zhang [aut], Jun Chen [aut, cre]

**Maintainer** Jun Chen <chen.jun2@mayo.edu>

**Repository** https://cran.r-universe.dev

**Date/Publication** 2025-05-13 08:50:06 UTC

**RemoteUrl** https://github.com/cran/SingleCellStat

**RemoteRef** HEAD

**RemoteSha** 1c4b455425ce92dc79af099aebf0300da4ca826d

## Contents

DiSC *DiSC: A statistical tool for differential expression analyis of individual level single-cell RNA-Seq data*

## Description

A statistical tool for differential expression analyis of individual level single-cell RNA-Seq data

## Usage

```
DiSC(data.mat, cell.ind, metadata, outcome, covariates = NULL,
     cell.id = "cell_id",
     individual.id = "individual", perm.no = 999,
     features = c('prev', 'm', 'nzsd'), verbose = TRUE,
     sequencing.data = TRUE)
```

## Arguments

| | |
|---|---|
| `data.mat` | A data matrix for single-cell RNA sequencing data, or other single-cell data such as CyToF data. Rows - genes/features, columns - cells. Column names are cell ids. |
| `cell.ind` | A data frame of cell-individual relationship. It includes two columns for cell ids and individual ids. It links cell ids to individual ids. |
| `metadata` | A data frame of individual-level metadata. It includes a column for individual ids, a column for the outcome of interest and columns for other covariates if applicable. |
| `outcome` | A character string for the column name of the outcome variable in `metadata`. |
| `covariates` | A character string or vector of character strings for the covariates to be adjusted. Should be the column names in `metadata`. Default: `NULL`. |
| `cell.id` | A character string for the column name of cell ids in `cell.ind`. |
| `individual.id` | A character string for the column name of the individual ids in `cell.ind` and `metadata`. |
| `perm.no` | An integer, number of permutations used. Default: `999`. It can be reduced to `99` if adjusted P-values (false discovery rate) are the only interest. |
| `features` | Features of the distribution used to test for the differentially expressed genes. Choose from `"prev"` (logit(non-zero proportion), `"nzm"` (sqrt(non-zero mean)), `"nzsd"` (sqrt(non-zero standard deviation)), `"m"` (overall mean)), `"sd"` (overall standard deviation), `"nzm^1"`, `"nzsd^1"`, `"m^1"`, `"sd^1"` ( non-sqrt-transformed versions). Default: `"prev"`, `"m"` and `"nzsd"`. |
| `verbose` | Logical. Should the function print the processes? Default: `TRUE`. |
| `sequencing.data` | Logical. Is the data.mat a sequencing data matrix (i.e., count data)? If TRUE, the total sum scaling will be used to normalize the count data. The users can normalize/transform the data themselves by setting it to be FALSE. Default: `TRUE`. |

## Value

**call** How was the function called?

**R2** Description of R2

**F0** Description of F0

**RSS** Description of RSS

**df.model** Description of df.model

**df.residual** Description of df.residual

**coef.list** Description of coef.list

**p.raw** Raw, unadjusted P-values.

**p.adj.fdr** P-values which have been adjusted for false discovery rate.

**p.adj.fwer** P-values which have been adjusted for family-wise error rate.

## Author(s)

Jun Chen <<chen.jun2@mayo.edu>> and Lujun Zhang

## References

Zhang, L., Yang, L., Ren, Y., Zhang, S., Guan, W., & Chen, J. (Bioinformatics): DiSC: a Statistical Tool for Fast Differential Expression Analysis of Individual-level Single-cell RNA-seq Data.

## Examples

```
set.seed(seed = 1234556)
data(sim_data)

count_matrix <- sim_data$count_matrix
meta_cell <- sim_data$meta_cell
gene_index <- sim_data$gene_index
meta_ind <- sim_data$meta_ind

obj1 <- DiSC(data.mat = count_matrix, cell.ind = meta_cell,
             metadata = meta_ind, outcome = "phenotype",
             covariates = "RIN", cell.id = "cell_id",
             individual.id = "individual", perm.no = 999,
             features = c('prev', 'm', 'nzsd'), verbose = TRUE,
             sequencing.data = TRUE)
# Type I error (the nominal level: 0.05)
mean(obj1$p.raw[gene_index$EE_index] <= 0.05)
# True positive rate (based on raw P-values, the higher the better.)
mean(obj1$p.raw[gene_index$mean_index] <= 0.05)
mean(obj1$p.raw[gene_index$var_index] <= 0.05)
mean(obj1$p.raw[gene_index$mean_var_index] <= 0.05)
# False discovery rate (the nominal level: 0.10)
sum(obj1$p.adj.fdr[gene_index$EE_index] <= 0.10)/
  sum(obj1$p.adj.fdr <= 0.10)
# True positive rate (based on FDR-adjusted P-values, the higher the better.)
mean(obj1$p.adj.fdr[gene_index$mean_index] <= 0.10)
```

```
mean(obj1$p.adj.fdr[gene_index$var_index] <= 0.10)
mean(obj1$p.adj.fdr[gene_index$mean_var_index] <= 0.10)

# By default, DiSC normalizes the scRNA-seq data using TSS (total sum scaling),
# adjusted for log median sequencing depths
# Other user-specified normalization methods can also be used:
# log2 transformed, adjusted for log median sequencing depth
# data_mat_log <- log2(data_mat+1)
# inds <- unique(meta_cell[["individual"]])
# meta_ind <- meta_ind[base::match(inds, meta_ind[["individual"]]), ]
# data_mat <- count_matrix
# depth <- colSums(data_mat)
# cell.list <- list()
# for (ind in inds)
#   cell.list[[ind]] <- meta_cell[meta_cell$individual == ind, ][["cell_id"]]
# log_md_depth <- numeric(length = length(inds))
# names(log_md_depth) <- inds
# for(ind in inds)
#   log_md_depth[ind] <- log(median(depth[cell.list[[ind]]]))
# meta_ind$log_md_depth <- log_md_depth
# obj_log <-
#   DiSC(data.mat = data_mat_log, cell.ind = meta_cell,
#        metadata = meta_ind, outcome = "phenotype",
#        covariates = c("RIN", "log_md_depth"),
#        cell.id = "cell_id", individual.id = "individual",
#        perm.no = 999, verbose = FALSE,
#        sequencing.data = FALSE, # sequencing.data needs to be FALSE
#        features = c('prev', 'm', 'nzsd'))
# Size factor: DESeq2, adjusted for log median sequencing depths
# require(DESeq2)
# colData <- data.frame(condition = rep(meta_ind$phenotype, each = 375),
#                       row.names = colnames(data_mat))
# dds <- DESeq2::DESeqDataSetFromMatrix(countData = data_mat + 1,
                                        # avoid every gene contains at least one zero
#                                         colData = colData, design = ~ condition)
# dds <- DESeq2::estimateSizeFactors(dds)
# data_mat_des <- sweep(data_mat, 2, DESeq2::sizeFactors(dds), FUN = "/")
# obj_des <-
#   DiSC(data.mat = data_mat_des, cell.ind = meta_cell,
#        metadata = meta_ind, outcome = "phenotype",
#        covariates = c("RIN", "log_md_depth"),
#        cell.id = "cell_id", individual.id = "individual",
#        perm.no = 999, verbose = FALSE,
#        sequencing.data = FALSE, # sequencing.data needs to be FALSE
#        features = c('prev', 'm', 'nzsd'))
```

---

## sim_data *Simulated Single-cell Sequencing Dataset*

---

## Description

This dataset was generated in the "Generate Simulation Datasets" step in the Parametric_simulation.rmd (https://github.com/Lujun995/DiSC_Replication_Code)

## Usage

```
data("sim_data")
```

## Format

It contains 12 cases and 12 controls, each with 375 cell replicates. The read depths of each cell replicate are well-balanced. A covariate called RIN (RNA Integrity Number) at the individual level is included in the dataset.

The dataset comprises a total of 1,000 genes. The signal density was 15%, with differences in mean, variance, and mean+variance (each at 5%). The ground truth of differential/equally expression genes are indicated by `gene_index`, including `mean_index` (genes with a difference in mean), `var_index` (genes with a difference in variance), `mean_var_index` (genes with a difference in both mean and variance), `EE_index` (otherwise (to estimate type-I error)).

A list of elements:

`count_matrix` A numeric count matrix.

`meta_cell` A data.frame of the metadata at the cell level.

`meta_ind` A data.frame of the metadata at the individual level.

`gene_index` A list of 4 numeric vectors representing the ground truth of the IDs of the differentially or equally expressed genes.

## Source

Simulated in the "Generate Simulation Datasets" step in the Parametric_simulation.rmd (https://github.com/Lujun995/DiSC_Replication_Code)

## Examples

```
data(sim_data)
str(sim_data)
```

# Index