Package 'CellDEEP'

March 29, 2026

Type Package

Title Cell DiffErential Expression by Pooling ('CellDEEP')

Version 1.0.1

Description Pool cells together before running differentially expression (DE) analysis.
Tell 'CellDEEP' how many cells you want to pool together (which shall be determined by the overall cell number of data), then run DE analysis.
Cheng et al. (2026) <doi:10.64898/2026.03.09.710522>.

License GPL (>= 3)

Encoding UTF-8

LazyData true

RoxygenNote 7.3.2

Imports Seurat

Suggests knitr, rmarkdown, testthat (>= 3.2.3)

Config/testthat/edition 3

VignetteBuilder knitr

Depends R (>= 3.5)

NeedsCompilation no

Author Yiyi Cheng [aut, cre] (ORCID: <https://orcid.org/0009-0005-3329-6842>)

Maintainer Yiyi Cheng <2593244c@student.gla.ac.uk>

Repository CRAN

Date/Publication 2026-03-29 15:30:02 UTC

Contents

CellDEEP.Kmean 2
CellDEEP.Random 3
FindMarker.CellDEEP 4
prepare_data 6
return.DE 6
sim 7

Index **9**

`CellDEEP.Kmean` *K-means Based Cell Pooling for Seurat Objects*

Description

Pools cells into "pseudocells" by applying k-means clustering to PCA embeddings. This reduces data sparsity while maintaining the biological grouping of sample, cluster, and condition.

Usage

```
CellDEEP.Kmean(
  dataset,
  n_cells = 10,
  nstart = 100,
  assay_name = "RNA",
  readcounts = "mean",
  min_cells_per_subgroup = 25
)
```

Arguments

`dataset`	A Seurat object. Must have PCA reductions calculated.
`n_cells`	Integer. Target number of cells to pool into each pseudocell.
`nstart`	Integer. Number of random sets to start with in `kmeans`.
`assay_name`	Character. The assay to pull counts from (default "RNA").
`readcounts`	Character. Aggregation method: "mean" (rounded average), "sum", "10X" (mean * 10).
`min_cells_per_subgroup`	Integer. Minimum cells required in each sample-cluster subgroup to perform pooling (default 25).

Value

A new Seurat object where each "cell" is a pooled group of original cells.

Note

This function requires that PCA has already been run on the input `dataset`, as it uses the "pca" reduction for clustering.

Examples

```
data("sim")
pool_input <- prepare_data(
  sim,
  sample_id = "DonorID",
  group_id = "Status",
```

```
  cluster_id = "cluster_id"
)

pooled_kmean <- CellDEEP.Kmean(
  pool_input,
  readcounts = "sum",
  n_cells = 3,
  min_cells_per_subgroup = 1,
  assay_name = "RNA"
)
pooled_kmean
```

`CellDEEP.Random` *Random Cell Pooling for Seurat Objects*

Description

Pools cells into pseudocells by random selection within biological groups. Includes a minimum threshold filter of 25 cells per subgroup to ensure pooling quality.

Usage

```
CellDEEP.Random(
  dataset,
  n_cells = 10,
  assay_name = "RNA",
  min_cells_per_subgroup = 25,
  readcounts = "mean"
)
```

Arguments

`dataset`	A Seurat object.
`n_cells`	Integer. The number of cells to pool into each pseudocell.
`assay_name`	Character. The assay to use for counts (default "RNA").
`min_cells_per_subgroup`	Integer. Minimum cells required in each sample-cluster subgroup to perform pooling (default 25).
`readcounts`	Character. Method to aggregate counts: "sum" or "mean".

Value

A new Seurat object containing the aggregated pseudocells.

Note

Subgroups (sample-cluster combinations) with fewer than 25 cells are automatically skipped. The function also generates a DimPlot to visualize the random pooling across samples.

Examples

```
data("sim")
pool_input <- prepare_data(
  sim,
  sample_id = "DonorID",
  group_id = "Status",
  cluster_id = "cluster_id"
)

pooled_random <- CellDEEP.Random(
  pool_input,
  readcounts = "sum",
  n_cells = 3,
  min_cells_per_subgroup = 1,
  assay_name = "RNA"
)
pooled_random
```

`FindMarker.CellDEEP` *Differential Expression with Optional Cell Pooling*

Description

It can run Seurat DE directly or first aggregate cells into metacells using CellDEEP pooling.

Usage

```
FindMarker.CellDEEP(
  object,
  ident.1 = NULL,
  ident.2 = NULL,
  group.by = "group_id",
  sample_id = NULL,
  group_id = NULL,
  cluster_id = NULL,
  prepare = TRUE,
  test.use = "wilcox",
  Pool = TRUE,
  readcounts = "sum",
  n_cells = 10,
  assay = "RNA",
  min_cells_per_subgroup = 25,
  cell_selection = "kmean",
  name.only = TRUE,
  logfc.threshold = 0.25,
  min.pct = 0.01,
  p_cutoff = 0.05,
```

```
  full_list = FALSE,
  ...
)
```

Arguments

object	A Seurat object.
ident.1	Character. First identity group to compare.
ident.2	Character. Second identity group to compare.
group.by	Character. Metadata column used for grouping (default "group_id").
sample_id	Character. Input metadata column for sample IDs.
group_id	Character. Input metadata column for group IDs.
cluster_id	Character. Input metadata column for cluster IDs.
prepare	Logical. If TRUE, run prepare_data first.
test.use	Character. DE test to use.
Pool	Logical. If TRUE, perform CellDEEP pooling before DE (default TRUE).
readcounts	Character. Pool aggregation method: "sum", "mean", or "10X".
n_cells	Integer. Target number of cells per pool.
assay	Character. Assay to use (default "RNA").
min_cells_per_subgroup	Integer. Minimum cells in each sample-cluster subgroup required for pooling.
cell_selection	Character. Pooling strategy: "kmean" or "random".
name.only	Logical. If TRUE, return gene names only.
logfc.threshold	Numeric. Minimum log fold-change.
min.pct	Numeric. Minimum detection rate.
p_cutoff	Numeric. Adjusted p-value threshold.
full_list	Logical. If TRUE, return all genes regardless of p-value.
...	Additional arguments passed to Seurat::FindMarkers.

Value

A vector of gene names or a DE data.frame.

prepare_data *Standardize Seurat Metadata for CellDEEP*

Description

Standardizes metadata columns to `sample_id`, `group_id`, and `cluster_id` so CellDEEP functions can run consistently.

Usage

```
prepare_data(
  Subset.Seurat,
  assay = "RNA",
  sample_id,
  group_id,
  cluster_id,
  file_path = NULL
)
```

Arguments

`Subset.Seurat`	A Seurat object.
`assay`	Character. Assay to use (default `"RNA"`).
`sample_id`	Character. Metadata column name for sample IDs.
`group_id`	Character. Metadata column name for group IDs.
`cluster_id`	Character. Metadata column name for cluster IDs.
`file_path`	Character. Reserved for compatibility.

Value

A Seurat object with standardized metadata fields.

return.DE *Perform Differential Expression and Filter Results*

Description

A wrapper for `Seurat::FindMarkers` that simplifies the extraction of Differentially Expressed (DE) genes. It supports p-value filtering and can return either gene names or a full results table.

Usage

```
return.DE(
  dataset,
  test.use = "wilcox",
  DE.ident.1,
  DE.ident.2,
  DE.group,
  assay = "RNA",
  p_cutoff = 0.05,
  name.only = TRUE,
  logfc.threshold = 0.25,
  min.pct = 0.01,
  full_list = FALSE,
  ...
)
```

Arguments

`dataset`	A Seurat object.
`test.use`	Character. DE test to use (default `"wilcox"`).
`DE.ident.1`	Identifier(s) for the first group of cells.
`DE.ident.2`	Identifier(s) for the second group of cells.
`DE.group`	Character. Metadata column to group by.
`assay`	Character. Assay to use (default `"RNA"`).
`p_cutoff`	Numeric. Adjusted p-value threshold (default 0.05).
`name.only`	Logical. If TRUE, return gene names only.
`logfc.threshold`	Numeric. Minimum log fold change (default 0.1).
`min.pct`	Numeric. Minimum fraction of cells expressing a gene.
`full_list`	Logical. If TRUE, return all genes and skip p-value filter.
`...`	Extra arguments passed to `Seurat::FindMarkers`.

Value

A character vector of genes or a marker data.frame.

`sim` *Sample simulated cells from muscat package*

Description

A dataset containing 200 simulated cells(100 per group) for demonstrating CellDEEP functions. Can be found at doi:10.5281/zenodo.18863779

Usage

```
data(sim)
```

Format

A Seurat object

Source

simulated data with muscat package

Index

CellDEEP.Kmean, 2
CellDEEP.Random, 3

FindMarker.CellDEEP, 4

prepare_data, 6

return.DE, 6

sim, 7