

# Package: CIDER (via r-universe)

March 9, 2025

**Type** Package

**Title** Meta-Clustering for scRNA-Seq Integration and Evaluation

**Version** 0.99.4

**Maintainer** Zhiyuan Hu <zhiyuan.cheryl.hu@gmail.com>

**Description** A workflow of (a) meta-clustering based on inter-group similarity measures and (b) a ground-truth-free test metric to assess the biological correctness of integration in real datasets. See Hu Z, Ahmed A, Yau C (2021) <doi:10.1101/2021.03.29.437525> for more details.

**URL** <https://github.com/zhiyuan-hu-lab/CIDER>,  
<https://zhiyuan-hu-lab.github.io/CIDER/>

**BugReports** <https://github.com/zhiyuan-hu-lab/CIDER/issues>

**Imports** limma (>= 3.42.0), edgeR (>= 3.28.0), stats (>= 3.6.2),  
foreach (>= 1.4.7), Seurat (>= 3.1.0), utils (>= 3.6.2),  
pheatmap (>= 1.0.0), dbscan (>= 1.1-5), kernlab (>= 0.9-29),  
doParallel, igraph, parallel, graphics, ggplot2, viridis

**License** MIT + file LICENSE

**Encoding** UTF-8

**LazyData** true

**LazyDataCompression** xz

**RoxygenNote** 7.3.2

**Suggests** knitr, rmarkdown, testthat (>= 3.0.0), statmod (>= 1.2.2),  
cowplot

**VignetteBuilder** knitr

**Depends** R (>= 3.5.0)

**Config/testthat/edition** 3

**NeedsCompilation** no

**Author** Zhiyuan Hu [aut, cre]

(<<https://orcid.org/0000-0002-1688-6032>>), Christopher Yau  
[aut] (<<https://orcid.org/0000-0001-7615-8523>>), Ahmed Ahmed  
[aut]

**Repository** CRAN

**Date/Publication** 2025-02-07 09:50:14 UTC

**Config/pak/sysreqs** libglpk-dev make libicu-dev libpng-dev libxml2-dev  
libssl-dev python3 zlib1g-dev

## Contents

calculateDistMatOneModel . . . . .	2
downsampling . . . . .	3
estimateProb . . . . .	4
finalClustering . . . . .	5
gatherInitialClusters . . . . .	6
getDistMat . . . . .	7
getGroupFit . . . . .	8
getIDEr . . . . .	8
hdbscan.seurat . . . . .	10
initialClustering . . . . .	11
mergeInitialClusters . . . . .	12
pancreas_meta . . . . .	13
plotDistMat . . . . .	14
plotHeatmap . . . . .	14
plotNetwork . . . . .	15
scatterPlot . . . . .	16
<b>Index</b>	<b>17</b>

---

calculateDistMatOneModel

*Calculate Distance Matrix Using a Single Model*

---

## Description

This function computes a similarity matrix by utilising a single linear model for differential expression analysis.

## Usage

```
calculateDistMatOneModel(
  matrix,
  metadata,
  verbose = TRUE,
  method = "voom",
  additional.variate = NULL
)
```

**Arguments**

matrix	A count matrix with rows representing genes or features and columns representing samples or cells.
metadata	A data frame containing metadata corresponding to the samples or cells. Each row should match a column in matrix.
verbose	Logical. If TRUE, the function displays progress messages and a progress bar. The default is TRUE.
method	A character string specifying the method for differential expression analysis. Options are "voom" or "trend", with "trend" as the default.
additional.variate	A character vector of additional variates to include in the linear model for regression.

**Value**

A similarity matrix.

**See Also**

[getDistMat](#)

---

downsampling	<i>Downsampling Cells</i>
--------------	---------------------------

---

**Description**

Downsamples cells from each group for IDER-based similarity calculation.

**Usage**

```
downsampling(
  metadata,
  n.size = 35,
  seed = NULL,
  include = FALSE,
  replace = FALSE,
  lower.cutoff = 3
)
```

**Arguments**

metadata	A data frame containing at least two columns: one for group labels and one for batch information. Each row corresponds to a single cell. Required.
n.size	Numeric value specifying the number of cells to use in each group. Default is 35.

seed	Numeric value to set the random seed for sampling. Default is 12345.
include	Logical value indicating whether to include groups that have fewer cells than n.size. Default is FALSE.
replace	Logical value specifying whether to sample with replacement if a group is smaller than n.size. Default is FALSE.
lower.cutoff	Numeric value indicating the minimum group size required for inclusion. Default is 3.

### Value

A list of numeric indices (or cell names) for cells to be kept for downstream computation.

### Examples

```
# 'meta' is a data frame with columns 'label' and 'batch'
meta <- data.frame(
  label = c(rep("A", 40), rep("A", 35), rep("B", 20)),
  batch = c(rep("X", 40), rep("Y", 35), rep("X", 20))
)
keep_cells <- downsampling(meta, n.size = 35, seed = 12345)

# Display the selected indices
print(keep_cells)
```

---

estimateProb	<i>Estimate the Empirical Probability of Whether Two Set of Cells from Distinct Batches Belong to the Same Population</i>
--------------	---

---

### Description

This function computes the empirical probability that two sets of cells from distinct batches belong to the same population, based on the output of getIDER.

### Usage

```
estimateProb(
  seu,
  ider,
  batch.var = "Batch",
  n_size = 40,
  n.perm = 5,
  verbose = FALSE
)
```

**Arguments**

seu	A Seurat object.
ider	A list returned by the getIDER function.
batch.var	Character string specifying the metadata column that contains batch information. Default is "Batch".
n_size	Numeric value indicating the number of cells per group used to compute the similarity. Default is 40.
n.perm	Numeric value specifying the number of permutations to perform.
verbose	Logical. If TRUE, progress messages are printed. Default is FALSE.

**Value**

A Seurat object with additional columns for the IDER-based similarity and the empirical probability of rejection.

**See Also**

[hdbscan.seurat](#), [getIDER](#)

---

finalClustering

*Final Clustering Step for Meta-Clustering*

---

**Description**

This function merges initial clusters into final clusters based on the IDER similarity matrix.

**Usage**

```
finalClustering(
  seu,
  dist,
  cutree.by = "h",
  cutree.h = 0.45,
  cutree.k = 3,
  hc.method = "complete"
)
```

**Arguments**

seu	A Seurat object that has undergone the getIDER step. Required.
dist	A list output from the getIDER function. Required.
cutree.by	Character string specifying whether to cut the dendrogram by height ("h") or by a fixed number of clusters ("k"). Default is "h".
cutree.h	Numeric value between 0 and 1 indicating the height at which to cut the dendrogram. This parameter is ignored if cutree.by = "k". Default is 0.45.

<code>cutree.k</code>	Numeric value specifying the number of clusters to generate if <code>cutree.by = "k"</code> . This parameter is ignored if <code>cutree.by = "h"</code> . Default is 3.
<code>hc.method</code>	Character string specifying the method to be used in hierarchical clustering (passed to <code>hclust</code> ).

**Value**

A Seurat object with the final clustering results stored in the `CIDER_clusters` column of its `meta.data`.

**See Also**

[getIDEr](#)

---

`gatherInitialClusters` *Gather Initial Cluster Names*

---

**Description**

Merge initial clustering results from a list of Seurat objects into a single Seurat object.

**Usage**

```
gatherInitialClusters(seu_list, seu)
```

**Arguments**

<code>seu_list</code>	A list containing Seurat objects with initial clustering results. Required.
<code>seu</code>	A Seurat object to which the merged initial cluster information will be added.

**Value**

A Seurat object containing the initial clustering results in the `initial_cluster` column of its `meta.data`.

**See Also**

[mergeInitialClusters](#)

---

`getDistMat`*Calculate the Similarity Matrix*

---

## Description

Compute the IDER-based similarity matrix for a list of Seurat objects. This function does not regress out batch effects and is designed for use during the initial clustering step.

## Usage

```
getDistMat(  
  seu_list,  
  verbose = TRUE,  
  tmp.initial.clusters = "seurat_clusters",  
  method = "trend",  
  batch.var = "Batch",  
  additional.variate = NULL,  
  downsampling.size = 35,  
  downsampling.include = TRUE,  
  downsampling.replace = TRUE  
)
```

## Arguments

<code>seu_list</code>	A list containing Seurat objects. Required.
<code>verbose</code>	Logical. If TRUE, progress messages and a progress bar are displayed. Default is TRUE.
<code>tmp.initial.clusters</code>	Character string specifying one of the column names from <code>Seurat@meta.data</code> that denotes groups, e.g., initial clusters. Default is "seurat_clusters".
<code>method</code>	Character string specifying the method for differential expression analysis. Options are "voom" or "trend" (default is "trend").
<code>batch.var</code>	Character string specifying the metadata column containing batch information. Default is "Batch".
<code>additional.variate</code>	Character vector of additional variates to include in the linear model for regression.
<code>downsampling.size</code>	Numeric value indicating the number of cells to use per group. Default is 35.
<code>downsampling.include</code>	Logical. Whether to include groups with fewer cells than <code>downsampling.size</code> . Default is TRUE.
<code>downsampling.replace</code>	Logical. Whether to sample with replacement for groups smaller than <code>downsampling.size</code> . Default is TRUE.

**Value**

A list of similarity matrices.

**See Also**

[calculateDistMatOneModel](#)

---

getGroupFit	<i>Calculate IDER-Based Similarity Between Two Groups</i>
-------------	---

---

**Description**

This function calculates the IDER-based similarity between two groups using a linear model.

**Usage**

```
getGroupFit(logCPM, design, contrast_m)
```

**Arguments**

logCPM	A numeric matrix of log-transformed counts per million.
design	A design matrix for the differential expression analysis.
contrast_m	A contrast matrix specifying the comparison between the two groups.

**Value**

A numeric value representing the IDER-based similarity between the two groups.

---

getIDER	<i>Compute IDER-Based Similarity</i>
---------	--------------------------------------

---

**Description**

Calculate the similarity matrix based on Inter-group Differential Expression (IDER) metrics with the selected batch effects regressed out.

**Usage**

```
getIDER(
  seu,
  group.by.var = "initial_cluster",
  batch.by.var = "Batch",
  verbose = TRUE,
  use.parallel = FALSE,
  n.cores = 1,
  downsampling.size = 40,
  downsampling.include = TRUE,
  downsampling.replace = TRUE
)
```

**Arguments**

seu	A Seurat S4 object that includes an <code>initial_cluster</code> column in its <code>meta.data</code> . Required.
group.by.var	Character string specifying the column in <code>seu@meta.data</code> that defines initial clusters (batch-specific groups). Default is "initial_cluster".
batch.by.var	Character string specifying the metadata column that indicates batch information. Default is "Batch".
verbose	Logical. If TRUE, progress messages and a progress bar are displayed. Default is TRUE.
use.parallel	Logical. If TRUE, parallel computation is used (requires <code>doParallel</code> ); in this case, no progress bar will be shown. Default is FALSE.
n.cores	Numeric. The number of cores to use for parallel computing. Default is 1.
downsampling.size	Numeric. The number of cells representing each group. Default is 40.
downsampling.include	Logical. Whether to include groups with fewer cells than <code>downsampling.size</code> . Default is FALSE.
downsampling.replace	Logical. Whether to sample with replacement if a group is smaller than <code>downsampling.size</code> . Default is FALSE.

**Value**

A list of objects: a similarity matrix, a numeric vector recording the cells used, and a data frame of the group combinations included.

**See Also**

[plotNetwork](#), [finalClustering](#)

---

`hdbscan.seurat`*Initial Clustering for Evaluating Integration*

---

### Description

This function applies HDBSCAN, a density-based clustering algorithm, to the corrected dimension reduction of a Seurat object.

### Usage

```
hdbscan.seurat(  
  seu,  
  batch.var = "Batch",  
  reduction = "pca",  
  dims = seq_len(15),  
  minPts = 25  
)
```

### Arguments

<code>seu</code>	A Seurat object containing integrated or batch-corrected data (e.g. PCA results).
<code>batch.var</code>	Character string specifying the metadata column that contains batch information. Default is "Batch".
<code>reduction</code>	Character string specifying the name of the dimension reduction to use (e.g. "PCA"). Default is "PCA".
<code>dims</code>	Numeric vector indicating the dimensions to be used for initial clustering. Default is 1:15.
<code>minPts</code>	Integer specifying the minimum number of points required to form a cluster. This value is passed to the <code>hdbscan</code> function. Default is 25.

### Value

A Seurat object with two additional columns in its `meta.data`: `dbscan_cluster` and `initial_cluster`.

### See Also

[getIDEr](#), [estimateProb](#)

---

initialClustering      *Initial Clustering*

---

### Description

Perform batch-specific initial clustering on a Seurat object.

### Usage

```
initialClustering(
  seu,
  batch.var = "Batch",
  cut.height = 0.4,
  nfeatures = 2000,
  additional.vars.to.regress = NULL,
  dims = seq_len(14),
  resolution = 0.6,
  downsampling.size = 50,
  verbose = FALSE
)
```

### Arguments

seu	A Seurat object. Required.
batch.var	Character string specifying one of the column names in <code>seu@meta.data</code> used to partition the object into subsets. Default is "Batch".
cut.height	Numeric value specifying the height at which to cut hierarchical trees. Default is 0.4.
nfeatures	Numeric value indicating the number of high-variance genes to use. Default is 2000.
additional.vars.to.regress	Character vector of additional variable names from <code>seu@meta.data</code> to regress out. Optional. Default is NULL.
dims	Numeric vector specifying the dimensions to be used for clustering (passed to Seurat). Default is 1:14.
resolution	Numeric value for clustering resolution (passed to Seurat). Default is 0.6.
downsampling.size	Numeric value indicating the number of cells representing each group. Default is 40.
verbose	Logical. If TRUE, a progress bar is displayed. Default is FALSE.

### Value

A Seurat S4 object with initial cluster assignments stored in the `initial_cluster` column of its `meta.data`.

**See Also**

[getIDEr](#), [finalClustering](#)

---

mergeInitialClusters    *Merge Initial Clusters*

---

**Description**

Merge initial clusters based on a provided similarity matrix and hierarchical clustering.

**Usage**

```
mergeInitialClusters(
  seu_list,
  dist_list,
  use = "coef",
  method = "hc",
  hc.method = "average",
  cutree.by = "h",
  cutree.h = 0.6,
  cutree.k = 3,
  batch.var = "Batch"
)
```

**Arguments**

seu_list	A list of Seurat objects containing the single-cell data. This parameter is required.
dist_list	A list of similarity matrices as returned by <code>getDistMat()</code> . The order of matrices should correspond to that of the Seurat objects in <code>seu_list</code> .
use	A string specifying the similarity measure to use. Currently, only "coef" is supported. Default is "coef".
method	A string specifying the clustering method to employ. The default is "hc" for hierarchical clustering.
hc.method	A string passed to the <code>method</code> parameter of <code>hclust()</code> . Default is "average".
cutree.by	A character indicating whether to cut the dendrogram by height ("h", default) or by a set number of clusters ("k").
cutree.h	A numeric value defining the height at which to cut the tree if <code>cutree.by = "h"</code> . Default is 0.6.
cutree.k	A numeric value specifying the number of clusters to generate if <code>cutree.by = "k"</code> . Default is 3.
batch.var	A character string representing the metadata column name that contains batch information. Default is "Batch".

### Details

This function accepts a list of Seurat objects and a corresponding list of similarity matrices, and then merges the initial clusters using a hierarchical clustering approach. The updated cluster assignments are stored within each Seurat object.

### Value

A list of Seurat objects in which the initial clustering has been updated. The new cluster assignments are stored in the `inicluster` field of each Seurat object, whilst the original assignments are preserved in the `inicluster_tmp` field.

### See Also

[gatherInitialClusters](#), [initialClustering](#)

---

pancreas\_meta

*Pancreas Metadata*

---

### Description

This dataset provides cell-level metadata for the human and mouse pancreatic data used in the study.

### Usage

```
data(pancreas_meta)
```

### Format

A data frame with 10127 rows and 3 columns:

**Batch** Species information (human or mouse).

**Group** Cell type annotation.

**Sample** Donor information.

### Details

Cell-level metadata for cross-species pancreatic data.

### Source

The metadata were downloaded alongside the count matrix from NCBI GEO accession GSE84133. Reference: Baron M, Veres A, Wolock SL, Faust AL, Gaujoux R, Vetere A, et al. A single-cell transcriptomic map of the human and mouse pancreas reveals inter- and intra-cell population structure. *Cell Syst.* 2016;3:346–360.e4.

---

plotDistMat	<i>Plot Similarity Matrix with pheatmap</i>
-------------	---

---

**Description**

This function creates a heatmap of the similarity matrix computed by `getDistMat()`.

**Usage**

```
plotDistMat(dist.list, use = "coef")
```

**Arguments**

<code>dist.list</code>	A list representing the similarity matrix output by <code>getDistMat()</code> . Required.
<code>use</code>	Character string specifying the similarity measure to use. Default is "coef". No other option is currently available.

**Value**

A pheatmap object displaying the similarity matrix.

**See Also**

[getDistMat](#)

---

plotHeatmap	<i>Plot Heatmap for the IDER-Based Similarity Matrix</i>
-------------	--

---

**Description**

This function generates a heatmap that visualises the similarity between shared groups across batches, as computed by `getIDER`.

**Usage**

```
plotHeatmap(seu, ider, batch.var = "Batch")
```

**Arguments**

<code>seu</code>	A Seurat object.
<code>ider</code>	The output list from the <code>getIDER</code> function.
<code>batch.var</code>	Character string specifying the metadata column that contains batch information. Default is "Batch".

**Value**

A heatmap displaying the similarity between shared groups across batches.

**See Also**

[getIDER](#)

---

plotNetwork

*Plot Network Graph*

---

**Description**

Visualise the network based on an IDER-based similarity matrix. The vertexes are initial clusters, and the edge width denotes the similarity between two initial clusters.

**Usage**

```
plotNetwork(
  seu,
  ider,
  batch.var = "Batch",
  colour.by = NULL,
  weight.factor = 6.5,
  col.vector = NULL,
  vertex.size = 1
)
```

**Arguments**

seu	Seurat S4 object after the step of getIDER, containing initial_cluster and Batch in its meta.data. Required.
ider	A list. Output of 'getIDER'. Required.
batch.var	Character. Metadata colname containing batch information. (Default: Batch)
colour.by	Character. It should be one of the colnames of Seurat object meta.data. It is used to colour the vertex of the network graph. (Default: NULL)
weight.factor	Numerical. Adjust the thickness of the edges. (Default: 6.5)
col.vector	A vector of Hex colour codes. If no value is given (default), a vector of 74 colours will be used.
vertex.size	Numerical. Adjust the size of vertexes. (Default: 1)

**Value**

An igraph object

**See Also**

[getIDER](#)

---

scatterPlot

*Scatterplot by a selected feature*


---

### Description

Scatterplot of a Seurat object based on dimension reduction.

### Usage

```
scatterPlot(
  seu,
  reduction,
  colour.by,
  colvec = NULL,
  title = NULL,
  sort.by.numbers = TRUE,
  viridis_option = "B"
)
```

### Arguments

seu	Seurat S4 object after the step of getIDER. Required.
reduction	Character. The dimension reduction used to plot. Common options: "pca", "tsne", "umap". The availability of dimension reduction can be checked by Reductions(seu).
colour.by	Character. One of the column names of seu@meta.data. Can be either discreet or continuous variables.
colvec	A vector of Hex colour codes. If no value is given (default), a vector of 74 colours will be used.
title	Character. Title of the figure.
sort.by.numbers	Boolean. Whether to sort the groups by the number of cells.(Default: True)
viridis_option	viridis_option. (Default: B)

### Value

A ggplot2 scatter plot

# Index

## \* datasets

- pancreas\_meta, [13](#)
- calculateDistMatOneModel, [2](#), [8](#)
- downsampling, [3](#)
- estimateProb, [4](#), [10](#)
- finalClustering, [5](#), [9](#), [12](#)
- gatherInitialClusters, [6](#), [13](#)
- getDistMat, [3](#), [7](#), [14](#)
- getGroupFit, [8](#)
- getIDEr, [5](#), [6](#), [8](#), [10](#), [12](#), [15](#)
- hdbscan.seurat, [5](#), [10](#)
- initialClustering, [11](#), [13](#)
- mergeInitialClusters, [6](#), [12](#)
- pancreas\_meta, [13](#)
- plotDistMat, [14](#)
- plotHeatmap, [14](#)
- plotNetwork, [9](#), [15](#)
- scatterPlot, [16](#)