

# Package ‘SIMLR’

December 19, 2024

**Version** 1.33.0

**Date** 2024-09-20

**Title** Single-cell Interpretation via Multi-kernel LeaRning (SIMLR)

**Depends** R (>= 4.1.0),

**Imports** parallel, Matrix, stats, methods, Rcpp, pracma, RcppAnnoy,  
RSpectra

**Suggests** BiocGenerics, BiocStyle, testthat, knitr, igraph

**Description** Single-cell RNA-seq technologies enable high throughput gene expression measurement of individual cells, and allow the discovery of heterogeneity within cell populations. Measurement of cell-to-cell gene expression similarity is critical for the identification, visualization and analysis of cell populations. However, single-cell data introduce challenges to conventional measures of gene expression similarity because of the high level of noise, outliers and dropouts. We develop a novel similarity-learning framework, SIMLR (Single-cell Interpretation via Multi-kernel LeaRning), which learns an appropriate distance metric from the data for dimension reduction, clustering and visualization.

**Encoding** UTF-8

**License** file LICENSE

**URL** <https://github.com/BatzoglouLabSU/SIMLR>

**BugReports** <https://github.com/BatzoglouLabSU/SIMLR>

**biocViews** ImmunoOncology, Clustering, GeneExpression, Sequencing,  
SingleCell

**RoxygenNote** 7.3.2

**LinkingTo** Rcpp

**NeedsCompilation** yes

**VignetteBuilder** knitr

**git\_url** <https://git.bioconductor.org/packages/SIMLR>

**git\_branch** devel

**git\_last\_commit** 5ff2f17

**git\_last\_commit\_date** 2024-10-29

**Repository** Bioconductor 3.21

**Date/Publication** 2024-12-18

**Author** Daniele Ramazzotti [aut] (ORCID:  
<https://orcid.org/0000-0002-6087-2666>),  
 Bo Wang [aut],  
 Luca De Sano [cre, aut] (ORCID:  
<https://orcid.org/0000-0002-9618-3774>),  
 Serafim Batzoglou [ctb]

**Maintainer** Luca De Sano <luca.desano@gmail.com>

## Contents

BuettnerFlorian . . . . .	2
SIMLR . . . . .	3
SIMLR_Estimate_Number_of_Clusters . . . . .	4
SIMLR_Feature_Ranking . . . . .	5
SIMLR_Large_Scale . . . . .	5
ZeiselAmit . . . . .	6
<b>Index</b>	<b>7</b>

---

BuettnerFlorian	<i>test dataset for SIMLR</i>
-----------------	-------------------------------

---

## Description

example dataset to test SIMLR from the work by Buettner, Florian, et al.

## Usage

```
data(BuettnerFlorian)
```

## Format

gene expression measurements of individual cells

## Value

list of 6: `in_X` = input dataset as an (m x n) gene expression measurements of individual cells, `n_clust` = number of clusters (number of distinct true labels), `true_labs` = ground true of cluster assignments for each of the `n_clust` clusters, `seed` = seed used to compute the results for the example, `results` = result by SIMLR for the inputs defined as described, `nmi` = normalized mutual information as a measure of the inferred clusters compared to the true labels

**Source**

Buettner, Florian, et al. "Computational analysis of cell-to-cell heterogeneity in single-cell RNA-sequencing data reveals hidden subpopulations of cells." *Nature biotechnology* 33.2 (2015): 155-160.

SIMLR

*SIMLR***Description**

perform the SIMLR clustering algorithm

**Usage**

```
SIMLR(
  X,
  c,
  no.dim = NA,
  k = 10,
  if.impute = FALSE,
  normalize = FALSE,
  cores.ratio = 1
)
```

**Arguments**

X	an (m x n) data matrix of gene expression measurements of individual cells or and object of class SCESet
c	number of clusters to be estimated over X
no.dim	number of dimensions
k	tuning parameter
if.impute	should I transpose the input data?
normalize	should I normalize the input data?
cores.ratio	ratio of the number of cores to be used when computing the multi-kernel

**Value**

clusters the cells based on SIMLR and their similarities

list of 8 elements describing the clusters obtained by SIMLR, of which y are the resulting clusters: y = results of k-means clusterings, S = similarities computed by SIMLR, F = results from network diffusion, ydata = data referring the the results by k-means, alphaK = clustering coefficients, execution.time = execution time of the present run, converge = iterative convergence values by T-SNE, LF = parameters of the clustering

**Examples**

```
data(BuettnerFlorian)
SIMLR(X = BuettnerFlorian$in_X, c = BuettnerFlorian$n_clust, cores.ratio = 0)
```

---

SIMLR\_Estimate\_Number\_of\_Clusters  
*SIMLR Estimate Number of Clusters*

---

**Description**

estimate the number of clusters by means of two huristics as discussed in the SIMLR paper

**Usage**

```
SIMLR_Estimate_Number_of_Clusters(X, NUMC = 2:5, cores.ratio = 1)
```

**Arguments**

<code>X</code>	an (m x n) data matrix of gene expression measurements of individual cells
<code>NUMC</code>	vector of number of clusters to be considered
<code>cores.ratio</code>	ratio of the number of cores to be used when computing the multi-kernel

**Value**

a list of 2 elements: K1 and K2 with an estimation of the best clusters (the lower values the better) as discussed in the original paper of SIMLR

**Examples**

```
data(BuettnerFlorian)
SIMLR_Estimate_Number_of_Clusters(BuettnerFlorian$in_X,
  NUMC = 2:5,
  cores.ratio = 0)
```

---

SIMLR\_Feature\_Ranking *SIMLR Feature Ranking*

---

**Description**

perform the SIMLR feature ranking algorithm. This takes as input the original input data and the corresponding similarity matrix computed by SIMLR

**Usage**

```
SIMLR_Feature_Ranking(A, X)
```

**Arguments**

A	an (n x n) similarity matrix by SIMLR
X	an (m x n) data matrix of gene expression measurements of individual cells

**Value**

a list of 2 elements: pvalues and ranking ordering over the n covariates as estimated by the method

**Examples**

```
data(BuettnerFlorian)
SIMLR_Feature_Ranking(A = BuettnerFlorian$results$S, X = BuettnerFlorian$in_X)
```

---

SIMLR\_Large\_Scale *SIMLR Large Scale*

---

**Description**

perform the SIMLR clustering algorithm for large scale datasets

**Usage**

```
SIMLR_Large_Scale(X, c, k = 10, kk = 100, if.impute = FALSE, normalize = FALSE)
```

**Arguments**

X	an (m x n) data matrix of gene expression measurements of individual cells or and object of class SCESet
c	number of clusters to be estimated over X
k	tuning parameter
kk	number of principal components to be assessed in the PCA
if.impute	should I transpose the input data?
normalize	should I normalize the input data?

**Value**

clusters the cells based on SIMLR Large Scale and their similarities

list of 8 elements describing the clusters obtained by SIMLR, of which y are the resulting clusters:  
 y = results of k-means clusterings, S0 = similarities computed by SIMLR, F = results from the large scale iterative procedure, ydata = data referring the the results by k-means, alphaK = clustering coefficients, val = distances from the k-nearest neighbour search, ind = indeces from the k-nearest neighbour search, execution.time = execution time of the present run

**Examples**

```
data(ZeiselAmit)
resized = ZeiselAmit$in_X[, 1:340]

SIMLR_Large_Scale(X = resized, c = ZeiselAmit$n_clust, k = 5, kk = 5)
```

---

ZeiselAmit	<i>test dataset for SIMLR large scale</i>
------------	---

---

**Description**

example dataset to test SIMLR large scale. This is a reduced version of the dataset from the work by Zeisel, Amit, et al.

**Usage**

```
data(ZeiselAmit)
```

**Format**

gene expression measurements of individual cells

**Value**

list of 6: in\_X = input dataset as an (m x n) gene expression measurements of individual cells, n\_clust = number of clusters (number of distinct true labels), true\_labs = ground true of cluster assignments for each of the n\_clust clusters, seed = seed used to compute the results for the example, results = result by SIMLR for the inputs defined as described, nmi = normalized mutual information as a measure of the inferred clusters compared to the true labels

**Source**

Zeisel, Amit, et al. "Cell types in the mouse cortex and hippocampus revealed by single-cell RNA-seq." *Science* 347.6226 (2015): 1138-1142.

# Index

BuettnerFlorian, [2](#)

SIMLR, [3](#)

SIMLR\_Estimate\_Number\_of\_Clusters, [4](#)

SIMLR\_Feature\_Ranking, [5](#)

SIMLR\_Large\_Scale, [5](#)

ZeiselAmit, [6](#)