

# Package: Daseq (via r-universe)

October 10, 2026

**Type** Package

**Title** Detecting regions of differential abundance between scRNA-seq datasets

**Version** 1.0.0

**Author** Jun Zhao <jun.zhao@yale.edu>

**Maintainer** Jun Zhao <jun.zhao@yale.edu>

**Description** DA-seq is a multiscale approach for detecting DA subpopulations. In contrast to clustering based approaches, our method can detect DA subpopulations that do not form well separated clusters. For each cell, we compute a multiscale differential abundance score measure. These scores are based on the k nearest neighbors in the transcriptome space for various values of k.

**Encoding** UTF-8

**LazyData** true

**RoxygenNote** 6.1.1

**Imports** RANN, glmnet, caret, Seurat, e1071, reticulate, ggplot2, cowplot, scales, ggrepel

**Config/pak/sysreqs** cmake libglpk-dev make libicu-dev libpng-dev libuv1-dev libxml2-dev libssl-dev python3 zlib1g-dev

**Repository** <https://blaserlab.r-universe.dev>

**Date/Publication** 2021-04-05 15:31:22 UTC

**RemoteUrl** <https://github.com/KlugerLab/Daseq>

**RemoteRef** HEAD

**RemoteSha** b1f58e0028975dcbe3b38e76a01832d30401aaf6

## Contents

|                      |   |
|----------------------|---|
| addDAslot . . . . .  | 2 |
| Daseq . . . . .      | 3 |
| getDacells . . . . . | 3 |

|                              |           |
|------------------------------|-----------|
| getDAregion . . . . .        | 4         |
| plotCellLabel . . . . .      | 5         |
| plotCellScore . . . . .      | 6         |
| plotDAsite . . . . .         | 7         |
| runSTG . . . . .             | 7         |
| SeuratLocalMarkers . . . . . | 8         |
| SeuratMarkerFinder . . . . . | 9         |
| STGlocalMarkers . . . . .    | 9         |
| STGmarkerFinder . . . . .    | 10        |
| updateDAcells . . . . .      | 11        |
| X.2d.melanoma . . . . .      | 12        |
| X.label.info . . . . .       | 13        |
| X.label.melanoma . . . . .   | 13        |
| X.melanoma . . . . .         | 14        |
| <b>Index</b>                 | <b>15</b> |

---

|           |                    |
|-----------|--------------------|
| addDAslot | <i>Add DA slot</i> |
|-----------|--------------------|

---

**Description**

Add DA region information to the meta.data of a Seurat object

**Usage**

addDAslot(object, da.regions, da.slot = "da", set.ident = F)

**Arguments**

|            |                                                                                                         |
|------------|---------------------------------------------------------------------------------------------------------|
| object     | input Seurat object                                                                                     |
| da.regions | output from function getDAregion()                                                                      |
| da.slot    | character, variable name to put in Seurat meta.data, default "da"                                       |
| set.ident  | a logical value to indicate whether to set Idents of the Seurat object to DA information, default False |

**Value**

updated Seurat object

---

|       |                                                                                      |
|-------|--------------------------------------------------------------------------------------|
| DAseq | <i>DAseq: Detecting regions of differential abundance between scRNA-seq datasets</i> |
|-------|--------------------------------------------------------------------------------------|

---

### Description

DA-seq is a multiscale approach for detecting DA subpopulations. In contrast to clustering based approaches, our method can detect DA subpopulations that do not form well separated clusters. For each cell, we compute a multiscale differential abundance score measure. These scores are based on the k nearest neighbors in the transcriptome space for various values of k.

### Author(s)

Jun Zhao, Ariel Jaffe, Henry Li, Ofir Lindenbaum, Xiuyuan Cheng, Yuval Kluger

---

|            |                                                    |
|------------|----------------------------------------------------|
| getDAcells | <i>DA-seq Step 1 &amp; Step 2: select DA cells</i> |
|------------|----------------------------------------------------|

---

### Description

Step 1: compute a multiscale score measure for each cell of its k-nearest-neighborhood for multiple values of k. Step 2: train a logistic regression classifier based on the multiscale score measure and retain cells that may reside in DA regions.

### Usage

```
getDAcells(X, cell.labels, labels.1, labels.2, k.vector = NULL,
  save.knn = F, alpha = 0, k.folds = 10, n.runs = 5, n.rand = 2,
  pred.thres = NULL, do.plot = T, plot.embedding = NULL,
  size = 0.5)
```

### Arguments

|             |                                                                                      |
|-------------|--------------------------------------------------------------------------------------|
| X           | size N-by-p matrix, input merged dataset of interest after dimension reduction.      |
| cell.labels | size N character vector, labels for each input cell                                  |
| labels.1    | character vector, label name(s) that represent condition 1                           |
| labels.2    | character vector, label name(s) that represent condition 2                           |
| k.vector    | vector, k values to create the score vector                                          |
| save.knn    | a logical value to indicate whether to save computed kNN result, default False       |
| alpha       | numeric, elasticnet mixing parameter passed to glmnet(), default 0 (Ridge)           |
| k.folds     | integer, number of data splits used in the neural network, default 10                |
| n.runs      | integer, number of times to run the neural network to get the predictions, default 5 |

|                             |                                                                                                                         |
|-----------------------------|-------------------------------------------------------------------------------------------------------------------------|
| <code>n.rand</code>         | integer, number of random permutations to run, default 2                                                                |
| <code>pred.thres</code>     | length-2 vector, top and bottom threshold on DA measure, default NULL, select significant DA cells based on permutation |
| <code>do.plot</code>        | a logical value to indicate whether to return ggplot objects showing the results, default True                          |
| <code>plot.embedding</code> | size N-by-2 matrix, 2D embedding for the cells                                                                          |
| <code>size</code>           | cell size to use in the plot, default 0.5                                                                               |

### Value

a list of results

**cell.idx** index of cells used in DA calculation

**da.ratio** score vector for each cell

**da.pred** (mean) prediction from the logistic regression

**da.up** index for DA cells more abundant in condition of labels.2

**da.down** index for DA cells more abundant in condition of labels.1

**pred.plot** ggplot object showing the predictions of logistic regression on plot.embedding

**da.cells.plot** ggplot object highlighting cells of da.cell.idx on plot.embedding

---

|             |                                      |
|-------------|--------------------------------------|
| getDAregion | <i>DA-seq Step 3: get DA regions</i> |
|-------------|--------------------------------------|

---

### Description

Cluster the DA cells retained from Step 1 and Step 2 of DA-seq to obtain spatially coherent DA regions.

### Usage

```
getDAregion(X, da.cells, cell.labels, labels.1, labels.2,
  prune.SNN = 1/15, resolution = 0.05, group.singletons = F,
  min.cell = NULL, do.plot = T, plot.embedding = NULL, size = 0.5,
  do.label = F, ...)
```

### Arguments

|                          |                                                                                |
|--------------------------|--------------------------------------------------------------------------------|
| <code>X</code>           | size N-by-p matrix, input merged dataset of interest after dimension reduction |
| <code>da.cells</code>    | output from getDAcells() or updateDAcells()                                    |
| <code>cell.labels</code> | size N vector, labels for each input cell                                      |
| <code>labels.1</code>    | vector, label name(s) that represent condition 1                               |
| <code>labels.2</code>    | vector, label name(s) that represent condition 2                               |
| <code>prune.SNN</code>   | parameter for Seurat function FindNeighbors(), default 1/15                    |

|                  |                                                                                                                             |
|------------------|-----------------------------------------------------------------------------------------------------------------------------|
| resolution       | parameter for Seurat function FindClusters(), default 0.05                                                                  |
| group.singletons | parameter for Seurat function FindClusters(), default True                                                                  |
| min.cell         | integer, number of cells below which a DA region will be removed as outliers, default NULL, use minimum k value in k-vector |
| do.plot          | a logical value to indicate whether to return ggplot objects showing the results, default True                              |
| plot.embedding   | size N-by-2 matrix, 2D embedding for the cells                                                                              |
| size             | cell size to use in the plot, default 0.5                                                                                   |
| do.label         | a logical value to indicate whether to label each DA region with text, default False                                        |
| ...              | other parameters to pass to Seurat FindClusters()                                                                           |

**Value**

a list of results

**cell.idx** index of cells used in DA calculation

**da.region.label** DA region label for each cell from the whole dataset, '0' represents non-DA cells.

**DA.stat** a table showing DA score and p-value for each DA region

**da.region.plot** ggplot object showing DA regions on plot.embedding

---

|               |                         |
|---------------|-------------------------|
| plotCellLabel | <i>Plot cell labels</i> |
|---------------|-------------------------|

---

**Description**

Produce a ggplot object with cells on 2D embedding, colored by given labels of each cell.

**Usage**

```
plotCellLabel(X, label, cell.col = NULL, size = 0.5, alpha = 1,
  shape = 16, do.label = T, label.size = 4, label.repel = F,
  label.plot = NULL)
```

**Arguments**

|          |                                                                         |
|----------|-------------------------------------------------------------------------|
| X        | matrix, 2D embedding of each cell for the plot                          |
| label    | vector, label for each cell                                             |
| cell.col | string vector, color bar to use for cell labels, default ggplot default |
| size     | numeric, dot size for each cell, default 0.5                            |
| alpha    | numeric between 0 to 1, dot opacity for each cell, default 1            |
| do.label | a logical value indicating whether to add text to mark each cell label  |

|                          |                                                                                            |
|--------------------------|--------------------------------------------------------------------------------------------|
| <code>label.size</code>  | numeric, size of text labels, default 4                                                    |
| <code>label.repel</code> | a logical value indicating whether to repel the labels to avoid overlapping, default False |
| <code>label.plot</code>  | cell labels to add text annotations, default NULL, add text for all labels                 |

**Value**

a ggplot object

---

|                            |                                   |
|----------------------------|-----------------------------------|
| <code>plotCellScore</code> | <i>Plot a score for each cell</i> |
|----------------------------|-----------------------------------|

---

**Description**

Produce a ggplot object with cells on 2D embedding, colored by a given score for each cell.

**Usage**

```
plotCellScore(X, score, cell.col = c("blue", "white", "red"),
  size = 0.5, alpha = 1, shape = 16)
```

**Arguments**

|                       |                                                                              |
|-----------------------|------------------------------------------------------------------------------|
| <code>X</code>        | matrix, 2D embedding of each cell for the plot                               |
| <code>score</code>    | numeric vector, a single value to color each cell, continuous                |
| <code>cell.col</code> | string vector, color bar to use for "score", default c("blue","white","red") |
| <code>size</code>     | numeric, dot size for each cell, default 0.5                                 |
| <code>alpha</code>    | numeric between 0 to 1, dot opacity for each cell, default 1                 |

**Value**

a ggplot object

---

plotDAsite

*Plot da site*


---

### Description

Plot da site

### Usage

```
plotDAsite(X, site.list, size = 0.5, cols = NULL)
```

### Arguments

|           |                                                                       |
|-----------|-----------------------------------------------------------------------|
| X         | matrix, 2D embedding of each cell for the plot                        |
| site.list | list, a list of cell indices for each site to plot                    |
| size      | numeric, dot size for each cell, default 0.5                          |
| cols      | string vector, color bar to use for each site, default ggplot default |

---

runSTG

*Run STG*


---

### Description

Run STG to select a set of genes that separate cells with label.1 from label.2 (other labels)

### Usage

```
runSTG(X, X.labels, label.1, label.2 = NULL, lambda = 1.5,
       n.runs = 5, return.model = T, python.use = "/usr/bin/python",
       GPU = "")
```

### Arguments

|              |                                                                                                                                |
|--------------|--------------------------------------------------------------------------------------------------------------------------------|
| X            | matrix, normalized expression matrix of all cells in the dataset, genes are in rows, rownames must be gene names               |
| X.labels     | numeric vector, specify labels for each cell, must be 0 or 1                                                                   |
| label.1      | cell label to define markers for                                                                                               |
| label.2      | second cell label to for comparison, if NULL, use all other labels                                                             |
| lambda       | numeric, regularization parameter that weights the number of selected genes, a larger lambda leads to fewer genes, default 1.5 |
| n.runs       | integer, number of runs to run the model, default 5                                                                            |
| return.model | a logical value to indicate whether to return the actual model of STG                                                          |
| python.use   | character string, the Python to use, default "/usr/bin/python"                                                                 |
| GPU          | which GPU to use, default "", using CPU                                                                                        |

**Value**

a list of results:

**markers** a list of data.frame with markers for each DA region

**accuracy** a numeric vector showing mean accuracy for each DA region

**model** a list of model for each DA region, each model contains:

**model** the model of STG of the final run

**cells** cell names/indices used to train the model

**features** features used to train the model

**selected.features** the selected features of the final run

**pred** the linear prediction value for each cell from the model

---

SeuratLocalMarkers      *Find local markers*

---

**Description**

Use Seurat FindMarkers() function to identify genes that distinguish a DA region from its local neighborhood

**Usage**

```
SeuratLocalMarkers(object, da.slot = "da", da.region.to.run,
  cell.label.slot, cell.label.to.run, ...)
```

**Arguments**

|                   |                                                                                                                       |
|-------------------|-----------------------------------------------------------------------------------------------------------------------|
| object            | input Seurat object                                                                                                   |
| da.slot           | character, variable name that represents DA regions in Seurat meta.data, default "da"                                 |
| da.region.to.run  | numeric, which (single) DA region to find local markers for                                                           |
| cell.label.slot   | character, variable name that represents cell labeling information in Seurat meta.data to combine with DA information |
| cell.label.to.run | cell label(s) that represent the local neighborhood for the input DA region                                           |
| ...               | parameters passed to Seurat FindMarkers() function                                                                    |

**Value**

a data.frame of markers and statistics

---

|                    |                                                                       |
|--------------------|-----------------------------------------------------------------------|
| SeuratMarkerFinder | <i>DA-seq Step 4: Seurat marker finder to characterize DA regions</i> |
|--------------------|-----------------------------------------------------------------------|

---

**Description**

Use Seurat FindMarkers() function to identify characteristic genes for DA regions

**Usage**

```
SeuratMarkerFinder(object, da.slot = "da", da.regions.to.run = NULL,
...)
```

**Arguments**

- object           input Seurat object
- da.slot           character, variable name that represents DA regions in Seurat meta.data, default "da"
- da.regions.to.run       numeric (vector), which DA regions to find markers for, default is to run all regions
- ...               parameters passed to Seurat FindMarkers() function

**Value**

a list of markers for DA regions with statistics

---

|                 |                                                                                                                       |
|-----------------|-----------------------------------------------------------------------------------------------------------------------|
| STGlocalMarkers | <i>STG local markers Run STG to find a set of genes that separate a given DA region from a local subset of cells.</i> |
|-----------------|-----------------------------------------------------------------------------------------------------------------------|

---

**Description**

STG local markers Run STG to find a set of genes that separate a given DA region from a local subset of cells.

**Usage**

```
STGlocalMarkers(X, da.regions, da.region.to.run, cell.label.info,
cell.label.to.run, ...)
```

**Arguments**

|                                |                                                                                                                                |
|--------------------------------|--------------------------------------------------------------------------------------------------------------------------------|
| <code>X</code>                 | matrix, normalized expression matrix of all cells in the dataset, genes are in rows, rownames must be gene names               |
| <code>da.regions</code>        | output from the function <code>getDAregion()</code>                                                                            |
| <code>da.region.to.run</code>  | numeric, which (single) DA region to find local markers for                                                                    |
| <code>cell.label.info</code>   | vector, cell labeling information to select the local subset of cells to compare with input DA region                          |
| <code>cell.label.to.run</code> | cell label(s) to select from <code>cell.label.info</code> that represent the local neighborhood for the input DA region        |
| <code>lambda</code>            | numeric, regularization parameter that weights the number of selected genes, a larger lambda leads to fewer genes, default 1.5 |
| <code>n.runs</code>            | integer, number of runs to run the model, default 5                                                                            |
| <code>python.use</code>        | character string, the Python to use, default <code>"/usr/bin/python"</code>                                                    |
| <code>return.model</code>      | a logical value to indicate whether to return the actual model of STG                                                          |
| <code>GPU</code>               | which GPU to use, default "", using CPU                                                                                        |

**Value**

a list of results:

**markers** a list of data.frame with markers for each DA region

**accuracy** a numeric vector showing mean accuracy for each DA region

**model** a list of model for each DA region, each model contains:

**model** the model of STG of the final run

**features** features used to train the model

**selected.features** the selected features of the final run

**pred** the linear prediction value for each cell from the model

---

STGmarkerFinder

*DA-seq Step 4: STG feature selection*

---

**Description**

Use STG (stochastic gates) to select genes that separate each DA region from the rest of the cells. For a full description of the algorithm, see Y. Yamada, O. Lindenbaum, S. Negahban, and Y. Kluger. Feature selection using stochastic gates. arXiv preprint arXiv:1810.04247, 2018.

**Usage**

```
STGmarkerFinder(X, da.regions, da.regions.to.run = NULL, lambda = 1.5,
  n.runs = 5, return.model = T, python.use = "/usr/bin/python",
  GPU = "")
```

**Arguments**

|                                |                                                                                                                                |
|--------------------------------|--------------------------------------------------------------------------------------------------------------------------------|
| <code>X</code>                 | matrix, normalized expression matrix of all cells in the dataset, genes are in rows, rownames must be gene names               |
| <code>da.regions</code>        | output from the function <code>getDAregion()</code>                                                                            |
| <code>da.regions.to.run</code> | numeric (vector), which DA regions to run the marker finder, default is to run all regions                                     |
| <code>lambda</code>            | numeric, regularization parameter that weights the number of selected genes, a larger lambda leads to fewer genes, default 1.5 |
| <code>n.runs</code>            | integer, number of runs to run the model, default 5                                                                            |
| <code>return.model</code>      | a logical value to indicate whether to return the actual model of STG                                                          |
| <code>python.use</code>        | character string, the Python to use, default <code>"/usr/bin/python"</code>                                                    |
| <code>GPU</code>               | which GPU to use, default <code>""</code> , using CPU                                                                          |

**Value**

a list of results:

**da.markers** a list of data.frame with markers for each DA region

**accuracy** a numeric vector showing mean accuracy for each DA region

**model** a list of model for each DA region, each model contains:

**model** the model of STG of the final run

**features** features used to train the model

**selected.features** the selected features of the final run

**pred** the linear prediction value for each cell from the model

---

|               |                        |
|---------------|------------------------|
| updateDAcells | <i>Update DA cells</i> |
|---------------|------------------------|

---

**Description**

Use different threshold to select DA cells based on an output from `getDAcells()`.

**Usage**

```
updateDAcells(X, pred.thres = NULL, force.thres = F, alpha = NULL,
  k.folds = 10, n.runs = 10, cell.labels = NULL, labels.1 = NULL,
  labels.2 = NULL, do.plot = T, plot.embedding = NULL, size = 0.5)
```

**Arguments**

|                             |                                                                                                                               |
|-----------------------------|-------------------------------------------------------------------------------------------------------------------------------|
| <code>x</code>              | output from <code>getDAcells()</code>                                                                                         |
| <code>pred.thres</code>     | length-2 vector, top and bottom threshold on DA measure, default NULL, select significant DA cells based on permutation       |
| <code>force.thres</code>    | a logical value to indicate whether to forcefully use <code>pred.thres</code> without considering significance, default False |
| <code>alpha</code>          | set this parameter to not NULL to rerun Logistic regression                                                                   |
| <code>do.plot</code>        | a logical value to indicate whether to return ggplot objects showing the results, default True                                |
| <code>plot.embedding</code> | size N-by-2 matrix, 2D embedding for the cells                                                                                |
| <code>size</code>           | cell size to use in the plot, default 0.5                                                                                     |

**Value**

a list of results with updated DA cells

---

|                            |                                                |
|----------------------------|------------------------------------------------|
| <code>X.2d.melanoma</code> | <i>t-SNE embedding of the melanoma dataset</i> |
|----------------------------|------------------------------------------------|

---

**Description**

A matrix containing 2D t-SNE embedding of the melanoma dataset

**Usage**

`X.2d.melanoma`

**Format**

An object of class `matrix` with 16291 rows and 2 columns.

**Source**

<https://www.sciencedirect.com/science/article/pii/S0092867418313941> Sade-Feldman, Moshe, et al. (Cell. 2018)

---

|              |                                 |
|--------------|---------------------------------|
| X.label.info | <i>Sample label information</i> |
|--------------|---------------------------------|

---

**Description**

A dataset containing information of each sample, indicating whether this sample is a "responder" (R) or a "non-responder" (NR)

**Usage**

```
X.label.info
```

**Format**

a data.frame with 48 rows and 2 columns

**label** sample label, matching with labels in X.label.melanoma

**condition** condition of the sample label, either R or NR

**Source**

<https://www.sciencedirect.com/science/article/pii/S0092867418313941> Sade-Feldman, Moshe, et al. (Cell. 2018)

---

|                  |                                                   |
|------------------|---------------------------------------------------|
| X.label.melanoma | <i>Cell sample labels of the melanoma dataset</i> |
|------------------|---------------------------------------------------|

---

**Description**

A string vector with the length equal to number of cells, indicating sample labels of each cell: which sample each cell comes from

**Usage**

```
X.label.melanoma
```

**Format**

An object of class character of length 16291.

**Source**

<https://www.sciencedirect.com/science/article/pii/S0092867418313941> Sade-Feldman, Moshe, et al. (Cell. 2018)

---

*X.melanoma**Top 10 PCs of the melanoma dataset*

---

**Description**

A dataset containing the top 10 PCs (principal components) of the melanoma dataset

**Usage**

`X.melanoma`

**Format**

An object of class `matrix` with 16291 rows and 10 columns.

**Source**

<https://www.sciencedirect.com/science/article/pii/S0092867418313941> Sade-Feldman, Moshe, et al. (Cell. 2018)

# Index

## \* datasets

- X.2d.melanoma, [12](#)
- X.label.info, [13](#)
- X.label.melanoma, [13](#)
- X.melanoma, [14](#)

addDASlot, [2](#)

DASeq, [3](#)

DASeq-package (DASeq), [3](#)

getDASlots, [3](#)

getDASite, [4](#)

plotCellLabel, [5](#)

plotCellScore, [6](#)

plotDASite, [7](#)

runSTG, [7](#)

SeuratLocalMarkers, [8](#)

SeuratMarkerFinder, [9](#)

STGlocalMarkers, [9](#)

STGmarkerFinder, [10](#)

updateDASlots, [11](#)

X.2d.melanoma, [12](#)

X.label.info, [13](#)

X.label.melanoma, [13](#)

X.melanoma, [14](#)