

# Package ‘scLang’

May 26, 2026

**Type** Package

**Title** A unified language for interacting with Seurat and  
SingleCellExperiment

**Version** 1.1.3

**Description** scLang is a suite for package development for scRNA-seq analysis.  
It offers functions that can operate on both Seurat and  
SingleCellExperiment objects. These functions are primarily aimed to help  
developers build tools compatible with both types of input.

**License** MIT + file LICENSE

**Imports** dplyr, ggplot2, henna, methods, paletteer, rlang, S4Vectors,  
SeuratObject, SingleCellExperiment, stats, SummarizedExperiment

**Suggests** BiocStyle, knitr, qs2, rmarkdown, scater, scRNAseq, Seurat,  
testthat (>= 3.0.0)

**biocViews** Software, SingleCell, GeneExpression, Visualization

**Encoding** UTF-8

**URL** <https://github.com/andrei-stoica26/scLang>

**BugReports** <https://github.com/andrei-stoica26/scLang/issues>

**RoxygenNote** 7.3.3

**VignetteBuilder** knitr

**Config/testthat/edition** 3

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

**git\_branch** devel

**git\_last\_commit** af892fd

**git\_last\_commit\_date** 2026-05-07

**Repository** Bioconductor 3.24

**Date/Publication** 2026-05-25

**Author** Andrei-Florian Stoica [aut, cre] (ORCID:  
<<https://orcid.org/0000-0002-5253-0826>>)

**Maintainer** Andrei-Florian Stoica <[andreistoica@foxmail.com](mailto:andreistoica@foxmail.com)>

## Contents

scLang-package . . . . .	2
dimPlot . . . . .	3
dimredNames.default . . . . .	4
documentFun . . . . .	5
featurePlot . . . . .	6
metadataDF . . . . .	7
metadataNames . . . . .	8
pickFeature . . . . .	8
scCol . . . . .	9
scColCounts . . . . .	10
scColPairCounts . . . . .	10
scColPairPercs . . . . .	11
scColPairRatios . . . . .	11
scDimredMat.default . . . . .	12
scExpMat.default . . . . .	13
scGeneExp.default . . . . .	14
scPCAMat . . . . .	15
scUMAPMat . . . . .	15
violinPlot . . . . .	16
<b>Index</b>	<b>18</b>

---

scLang-package	<i>scLang: A unified language for interacting with Seurat and Single-CellExperiment</i>
----------------	---

---

## Description

scLang is a suite for package development for scRNA-seq analysis. It offers functions that can operate on both Seurat and SingleCellExperiment objects. These functions are primarily aimed to help developers build tools compatible with both types of input.

## Author(s)

**Maintainer:** Andrei-Florian Stoica <andreistoica@foxmail.com> ([ORCID](#))

## See Also

Useful links:

- <https://github.com/andrei-stoica26/scLang>
- Report bugs at <https://github.com/andrei-stoica26/scLang/issues>

---

<code>dimPlot</code>	<i>Creates a dimensionality reduction plot</i>
----------------------	--

---

### Description

This function creates a dimensionality reduction plot

### Usage

```
dimPlot(
  scObj,
  groupBy = NULL,
  title = NULL,
  dimred = "umap",
  dims = c(1, 2),
  legendTitle = "Group",
  noGroupsLegendLab = "Object",
  palette = "grDevices::rainbow",
  pointSize = 0.5,
  alpha = 0.7,
  legendPos = c("right", "top", "left", "bottom"),
  legendTextSize = 10,
  legendTitleSize = 10,
  axisTextSize = 12,
  axisTitleSize = 12,
  ...
)
```

### Arguments

<code>scObj</code>	A Seurat or SingleCellExperiment object.
<code>groupBy</code>	Grouping variable. Must exist in the metadata/coldata of the single-cell expression object.
<code>title</code>	Plot title.
<code>dimred</code>	Dimensionality reduction.
<code>dims</code>	A numeric vector of size 2 representing the dimensions selected for the plot.
<code>legendTitle</code>	Legend title.
<code>noGroupsLegendLab</code>	Legend label to be used when no grouping is provided (groupBy is NULL)
<code>palette</code>	Color palette.
<code>pointSize</code>	Point size.
<code>alpha</code>	Opacity level.
<code>legendPos</code>	Legend position.
<code>legendTextSize</code>	Legend text size.
<code>legendTitleSize</code>	Legend title size.
<code>axisTextSize</code>	Axis text size.
<code>axisTitleSize</code>	Axis title size.
<code>...</code>	Additional arguments passed to <code>henna::centerTitle</code> .

**Value**

A dimensionality reduction plot.

**Examples**

```
scePath <- system.file('extdata', 'sceObj.qs2', package='scLang')
sceObj <- qs2::qs_read(scePath)
dimPlot(sceObj, groupBy='Donor')
```

---

dimredNames.default     *Extract the names of available dimensionality reductions from object*

---

**Description**

This function extracts the names of available dimensionality reductions from a Seurat or SingleCellExperiment object.

**Usage**

```
## Default S3 method:
dimredNames(scObj)

## S3 method for class 'Seurat'
dimredNames(scObj)

## S3 method for class 'SingleCellExperiment'
dimredNames(scObj)

dimredNames(scObj)
```

**Arguments**

scObj                    A Seurat or SingleCellExperiment object.

**Value**

A character vector.

**Examples**

```
scePath <- system.file('extdata', 'sceObj.qs2', package='scLang')
sceObj <- qs2::qs_read(scePath)
dimredNames(sceObj)
```

---

`documentFun`*Internal function used for documenting other functions*

---

**Description**

This function is used internally to help document other functions.

**Usage**

```
documentFun(  
  scObj = NULL,  
  title = NULL,  
  groupBy = NULL,  
  dimred = "umap",  
  dims = c(1, 2),  
  xLab = "x",  
  yLab = "y",  
  legendTitle = "Legend",  
  legendLabs = c("a", "b"),  
  legendPos = "right",  
  palette = "Spectral",  
  alpha = 1,  
  pointSize = 0.8,  
  legendTitleSize = 10,  
  legendTextSize = 10,  
  axisTextSize = 12,  
  axisTitleSize = 12,  
  ...  
)
```

**Arguments**

<code>scObj</code>	A Seurat or SingleCellExperiment object.
<code>title</code>	Plot title.
<code>groupBy</code>	Grouping variable. Must exist in the metadata/coldata of the single-cell expression object.
<code>dimred</code>	Dimensionality reduction.
<code>dims</code>	A numeric vector of size 2 representing the dimensions selected for the plot.
<code>xLab</code>	x axis label.
<code>yLab</code>	y axis label.
<code>legendTitle</code>	Legend title.
<code>legendLabs</code>	Legend labels.
<code>legendPos</code>	Legend position.
<code>palette</code>	Color palette.
<code>alpha</code>	Opaqueness level.
<code>pointSize</code>	Point size.

```

legendTitleSize Legend title size.
legendTextSize  Legend text size.
axisTextSize    Axis text size.
axisTitleSize   Axis title size.
...             Additional arguments passed to henna::centerTitle.

```

**Value**

NULL. This function is only used internally for documentation.

---

<code>featurePlot</code>	<i>Create a dimensionality reduction plot to represent a feature</i>
--------------------------	--

---

**Description**

This function creates a dimensionality reduction plot to represent a feature (gene expression or numeric metadata column).

**Usage**

```

featurePlot(
  scObj,
  feature = rownames(scObj)[1],
  title = feature,
  dimred = "umap",
  dims = c(1, 2),
  legendTitle = NULL,
  palette = paletteer_d("wesanderson::Royal1")[c(3, 2)],
  pointSize = 0.5,
  alpha = 0.6,
  legendPos = c("right", "top", "left", "bottom"),
  legendTextSize = 10,
  legendTitleSize = 10,
  axisTextSize = 12,
  axisTitleSize = 12,
  ...
)

```

**Arguments**

<code>scObj</code>	A Seurat or SingleCellExperiment object.
<code>feature</code>	A gene name or metadata column name.
<code>title</code>	Plot title.
<code>dimred</code>	Dimensionality reduction.
<code>dims</code>	A numeric vector of size 2 representing the dimensions selected for the plot.
<code>legendTitle</code>	Legend title.
<code>palette</code>	Color palette.

pointSize	Point size.
alpha	Opaqueness level.
legendPos	Legend position.
legendTextSize	Legend text size.
legendTitleSize	Legend title size.
axisTextSize	Axis text size.
axisTitleSize	Axis title size.
...	Additional arguments passed to <code>henna::centerTitle</code> .

**Value**

A feature plot.

**Examples**

```
scePath <- system.file('extdata', 'sceObj.qs2', package='scLang')
sceObj <- qs2::qs_read(scePath)
featurePlot(sceObj, 'Gene289')
```

---

metadataDF

*Extract metadata from object as a data frame*

---

**Description**

This function extracts the metadata from a `Seurat` or `SingleCellExperiment` object as a data frame.

**Usage**

```
metadataDF(scObj)

## Default S3 method:
metadataDF(scObj)

## S3 method for class 'Seurat'
metadataDF(scObj)

## S3 method for class 'SingleCellExperiment'
metadataDF(scObj)
```

**Arguments**

`scObj` A `Seurat` or `SingleCellExperiment` object.

**Value**

A metadata data frame.

**Examples**

```
scePath <- system.file('extdata', 'sceObj.qs2', package='scLang')
sceObj <- qs2::qs_read(scePath)
df <- metadataDF(sceObj)
```

---

metadataNames	<i>Return metadata names</i>
---------------	------------------------------

---

**Description**

This function extracts metadata names from a Seurat or SingleCellExperiment object. It can also be used to modify metadata names.

**Usage**

```
metadataNames(scObj)
```

**Arguments**

scObj            A Seurat or SingleCellExperiment object.

**Value**

The names of the metadata columns.

**Examples**

```
scePath <- system.file('extdata', 'sceObj.qs2', package='scLang')
sceObj <- qs2::qs_read(scePath)
colNames <- metadataNames(sceObj)
```

---

pickFeature	<i>Extract feature data to be used by featurePlot or violinPlot</i>
-------------	---

---

**Description**

This function extracts feature data to be used by featurePlot or violinPlot.

**Usage**

```
pickFeature(scObj, feature)
```

**Arguments**

scObj            A Seurat or SingleCellExperiment object.  
feature          A gene name or metadata column name.

**Value**

A numeric vector.

---

scCol	<i>Extract a metadata/coldata column from object.</i>
-------	---

---

## Description

This function extracts a metadata/coldata column from a Seurat or SingleCellExperiment object.

## Usage

```
scCol(scObj, col)

scCol(scObj, col) <- value

## Default S3 method:
scCol(scObj, col)

## Default S3 replacement method:
scCol(scObj, col) <- value

## S3 method for class 'Seurat'
scCol(scObj, col)

## S3 replacement method for class 'Seurat'
scCol(scObj, col) <- value

## S3 method for class 'SingleCellExperiment'
scCol(scObj, col)

## S3 replacement method for class 'SingleCellExperiment'
scCol(scObj, col) <- value
```

## Arguments

scObj	A Seurat or SingleCellExperiment object.
col	Column name.
value	A vector to be added to the metadata/coldata of the single-cell expression object.

## Value

A vector.

## Examples

```
scePath <- system.file('extdata', 'sceObj.qs2', package='scLang')
sceObj <- qs2::qs_read(scePath)
v <- scCol(sceObj, 'Cluster')
```

---

scColCounts	<i>Extract per-group counts from the column of a single-cell expression object</i>
-------------	--

---

**Description**

This function extracts per-group counts from the column of single-cell expression object.

**Usage**

```
scColCounts(scObj, col)
```

**Arguments**

scObj	A Seurat or SingleCellExperiment object.
col	Column as string.

**Value**

A frequency vector with the unique column values as names.

**Examples**

```
scePath <- system.file('extdata', 'sceObj.qs2', package='scLang')
sceObj <- qs2::qs_read(scePath)
scColCounts(sceObj, 'Cluster')
```

---

scColPairCounts	<i>Extract counts from two columns of a single-cell expression object</i>
-----------------	---

---

**Description**

This function extracts count information from two columns of a single-cell expression object.

**Usage**

```
scColPairCounts(scObj, col1, col2)
```

**Arguments**

scObj	A Seurat or SingleCellExperiment object.
col1	Column as string.
col2	Column as string.

**Value**

A data frame listing the counts of all combinations of pairs from two categorical columns.

**Examples**

```
scePath <- system.file('extdata', 'sceObj.qs2', package='scLang')
sceObj <- qs2::qs_read(scePath)
scColPairCounts(sceObj, 'Cluster', 'Donor')
```

---

scColPairPercs	<i>Extract percentages from two columns of a single-cell expression object</i>
----------------	--

---

**Description**

This function extracts percentage information from two columns of a single-cell expression object. For each  $i \times j$  combination with  $i$  taken from column 1 and  $j$  taken from column 2, the function reports the percentage that  $j$  contributes to all combinations involving  $i$ .

**Usage**

```
scColPairPercs(scObj, col1, col2, sigDigits = 2)
```

**Arguments**

scObj	A Seurat or SingleCellExperiment object.
col1	Column as string.
col2	Column as string.
sigDigits	Number of significant digits.

**Value**

A data frame listing the percentages of all combinations of pairs from two categorical columns.

**Examples**

```
scePath <- system.file('extdata', 'sceObj.qs2', package='scLang')
sceObj <- qs2::qs_read(scePath)
scColPairPercs(sceObj, 'Cluster', 'Donor')
```

---

scColPairRatios	<i>Extract observed-over-expected ratios from two columns of a single-cell expression object</i>
-----------------	--

---

**Description**

This function extracts observed-over-expected ratios from two columns of a single-cell expression object.

**Usage**

```
scColPairRatios(scObj, col1, col2, sigDigits = 2)
```

**Arguments**

scObj            A Seurat or SingleCellExperiment object.  
col1            Column as string.  
col2            Column as string.  
sigDigits       Number of significant digits.

**Value**

A data frame listing the observed-over-expected ratios of all combinations of pairs from two categorical columns.

**Examples**

```
scePath <- system.file('extdata', 'sceObj.qs2', package='scLang')  
sceObj <- qs2::qs_read(scePath)  
scColPairRatios(sceObj, 'Cluster', 'Donor')
```

---

scDimredMat.default    *Extracts a dimensionality reduction matrix from object.*

---

**Description**

This function extracts a dimensionality reduction matrix from a Seurat or SingleCellExperiment object.

**Usage**

```
## Default S3 method:  
scDimredMat(scObj, dimred)  
  
## S3 method for class 'Seurat'  
scDimredMat(scObj, dimred)  
  
## S3 method for class 'SingleCellExperiment'  
scDimredMat(scObj, dimred)  
  
scDimredMat(scObj, dimred)
```

**Arguments**

scObj            A Seurat or SingleCellExperiment object.  
dimred           Dimensionality reduction.

**Value**

A dimensionality reduction matrix.

**Examples**

```
scePath <- system.file('extdata', 'sceObj.qs2', package='scLang')
sceObj <- qs2::qs_read(scePath)
pcaMat <- scDimredMat(sceObj, 'pca')
```

---

scExpMat.default	<i>Extracts the expression matrix from object.</i>
------------------	--

---

**Description**

This function extracts an expression matrix from a Seurat or SingleCellExperiment object. For additional flexibility, the input can alternatively be provided as a dgMatrix object (and an option to densify it is also available) or as a matrix object.

**Usage**

```
## Default S3 method:
scExpMat(sceObj, slot = NULL, genes = NULL, densify = TRUE)

## S3 method for class 'Seurat'
scExpMat(sceObj, slot = "data", genes = NULL, densify = TRUE)

## S3 method for class 'SingleCellExperiment'
scExpMat(sceObj, slot = "logcounts", genes = NULL, densify = TRUE)

## S3 method for class 'dgMatrix'
scExpMat(sceObj, slot = NULL, genes = NULL, densify = TRUE)

## S3 method for class 'matrix'
scExpMat(sceObj, slot = NULL, genes = NULL, densify = TRUE)

scExpMat(sceObj, slot, genes = NULL, densify = TRUE)
```

**Arguments**

scObj	A Seurat, SingleCellExperiment, dgMatrix or matrix object.
slot	Gene expression slot. Ignored if scObj is of class dgMatrix or matrix.
genes	Selected genes. If NULL, all genes will be retained.
densify	Whether to convert to dense matrix.

**Value**

An expression matrix.

**Examples**

```
scePath <- system.file('extdata', 'sceObj.qs2', package='scLang')
sceObj <- qs2::qs_read(scePath)
mat <- scExpMat(sceObj, 'counts')
```

---

scGeneExp.default	<i>Extracts the expression of a single gene</i>
-------------------	---

---

### Description

This function extracts the expression of a single gene from a Seurat, SingleCellExperiment, dgCMatrx or matrix object.

### Usage

```
## Default S3 method:  
scGeneExp(scObj, gene, slot = NULL)  
  
## S3 method for class 'Seurat'  
scGeneExp(scObj, gene, slot = "data")  
  
## S3 method for class 'SingleCellExperiment'  
scGeneExp(scObj, gene, slot = "logcounts")  
  
## S3 method for class 'dgCMatrx'  
scGeneExp(scObj, gene, slot = NULL)  
  
## S3 method for class 'matrix'  
scGeneExp(scObj, gene, slot = NULL)  
  
scGeneExp(scObj, gene, slot)
```

### Arguments

scObj	A Seurat, SingleCellExperiment, dgCMatrx or matrix object.
gene	Selected gene.
slot	Gene expression slot. Ignored if scObj is of class dgCMatrx or matrix.

### Value

A gene expression vector.

### Examples

```
scePath <- system.file('extdata', 'sceObj.qs2', package='scLang')  
sceObj <- qs2::qs_read(scePath)  
v <- scGeneExp(sceObj, 'Gene291')
```

---

`scPCAMat`*Extracts the PCA matrix from object.*

---

**Description**

This function extracts the PCA matrix from a Seurat or SingleCellExperiment object.

**Usage**

```
scPCAMat(scObj)
```

**Arguments**

`scObj` A Seurat or SingleCellExperiment object.

**Value**

A PCA matrix.

**Examples**

```
scePath <- system.file('extdata', 'sceObj.qs2', package='scLang')
sceObj <- qs2::qs_read(scePath)
pcaMat <- scPCAMat(sceObj)
```

---

`scUMAPMat`*Extracts the UMAP matrix from object.*

---

**Description**

This function extracts the UMAP matrix from a Seurat or SingleCellExperiment object.

**Usage**

```
scUMAPMat(scObj)
```

**Arguments**

`scObj` A Seurat or SingleCellExperiment object.

**Value**

A UMAP matrix.

**Examples**

```
scePath <- system.file('extdata', 'sceObj.qs2', package='scLang')
sceObj <- qs2::qs_read(scePath)
umapMat <- scUMAPMat(sceObj)
```

---

violinPlot

*Create a violin plot to represent a feature*


---

### Description

This function creates a violin plot to represent a feature (gene expression or numeric metadata column).

### Usage

```
violinPlot(
  scObj,
  feature = rownames(scObj)[1],
  groupBy = metadataNames(scObj)[1],
  title = feature,
  legendTitle = NULL,
  xLab = "Identity",
  yLab = "Expression level",
  palette = "grDevices::rainbow",
  pointSize = 0.5,
  alpha = 0.8,
  legendPos = c("right", "top", "left", "bottom"),
  legendTextSize = 10,
  legendTitleSize = 10,
  axisTextSize = 12,
  axisTitleSize = 12,
  xLabAngle = 45,
  xLabVjust = 0.5,
  ...
)
```

### Arguments

scObj	A Seurat or SingleCellExperiment object.
feature	A gene name or metadata column name.
groupBy	Grouping variable. Must exist in the metadata/coldata of the single-cell expression object.
title	Plot title.
legendTitle	Legend title.
xLab	x axis label.
yLab	y axis label.
palette	Color palette.
pointSize	Point size.
alpha	Opacity level.
legendPos	Legend position.
legendTextSize	Legend text size.
legendTitleSize	Legend title size.

<code>axisTextSize</code>	Axis text size.
<code>axisTitleSize</code>	Axis title size.
<code>xLabAngle</code>	x axis label angle.
<code>xLabVjust</code>	x axis label vertical justification in [0, 1].
<code>...</code>	Additional arguments passed to <code>henna::centerTitle</code> .

**Value**

A violin plot.

**Examples**

```
scePath <- system.file('extdata', 'sceObj.qs2', package='scLang')
sceObj <- qs2::qs_read(scePath)
violinPlot(sceObj, 'Gene289')
```

# Index

## \* internal

- documentFun, 5
- pickFeature, 8
- scLang-package, 2

dimPlot, 3  
dimredNames (dimredNames.default), 4  
dimredNames.default, 4  
documentFun, 5

featurePlot, 6

metadataDF, 7  
metadataNames, 8

pickFeature, 8

scCol, 9  
scCol<- (scCol), 9  
scColCounts, 10  
scColPairCounts, 10  
scColPairPercs, 11  
scColPairRatios, 11  
scDimredMat (scDimredMat.default), 12  
scDimredMat.default, 12  
scExpMat (scExpMat.default), 13  
scExpMat.default, 13  
scGeneExp (scGeneExp.default), 14  
scGeneExp.default, 14  
scLang (scLang-package), 2  
scLang-package, 2  
scPCAMat, 15  
scUMAPMat, 15

violinPlot, 16