

# Package ‘sketchR’

April 4, 2025

**Version** 1.3.0

**Date** 2024-10-07

**Title** An R interface for python subsampling/sketching algorithms

**License** MIT + file LICENSE

**Description** Provides an R interface for various subsampling algorithms implemented in python packages. Currently, interfaces to the geosketch and scSampler python packages are implemented. In addition it also provides diagnostic plots to evaluate the subsampling.

**Imports** basilisk, Biobase, DelayedArray, dplyr, ggplot2, methods, reticulate, rlang, scales, stats

**Suggests** rmarkdown, knitr, testthat (>= 3.0.0), TENxPBMCDData, scuttle, scran, scater, SingleR, celldex, cowplot, SummarizedExperiment, beachmat.hdf5, BiocStyle, BiocManager, SingleCellExperiment

**URL** <https://github.com/fmicompbio/sketchR>

**BugReports** <https://github.com/fmicompbio/sketchR/issues>

**RoxygenNote** 7.3.2

**Encoding** UTF-8

**StagedInstall** no

**Config/testthat/edition** 3

**VignetteBuilder** knitr

**biocViews** SingleCell

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

**git\_branch** devel

**git\_last\_commit** e11ec9e

**git\_last\_commit\_date** 2024-12-21

**Repository** Bioconductor 3.21

**Date/Publication** 2025-04-03

**Author** Charlotte Soneson [aut, cre] (ORCID:  
 <<https://orcid.org/0000-0003-3833-2169>>),  
 Michael Stadler [aut] (ORCID: <<https://orcid.org/0000-0002-2269-4934>>),  
 Friedrich Miescher Institute for Biomedical Research [cph]

**Maintainer** Charlotte Soneson <charlottesoneson@gmail.com>

## Contents

sketchR-package . . . . .	2
compareCompositionPlot . . . . .	3
geosketch . . . . .	4
getGeosketchNames . . . . .	5
getScSamplerNames . . . . .	6
hausdorffDistPlot . . . . .	6
scsampler . . . . .	8
<b>Index</b>	<b>9</b>

---

sketchR-package	<i>sketchR - an R interface for python subsampling/sketching algorithms</i>
-----------------	---

---

## Description

The sketchR package provides an R interface for various subsampling algorithms implemented in python packages. Currently, interfaces to the geosketch and scSampler python packages are implemented, in the functions geosketch() and scsampler(), respectively. In addition the package also provides diagnostic plots to evaluate the subsampling. More details on how to get started and incorporate the subsampling into an scRNA-seq workflow are provided in the vignette.

## Author(s)

Charlotte Soneson  
 Michael Stadler

## See Also

Useful links:

- <https://github.com/fmicompbio/sketchR>
- Report bugs at <https://github.com/fmicompbio/sketchR/issues>

---

`compareCompositionPlot`*Compare the compositions of a data set and a subset*

---

**Description**

Plot the composition of a data set (e.g., the number of cells from each cell type) and contrast it with the corresponding composition of a subset.

**Usage**

```
compareCompositionPlot(  
  df,  
  idx,  
  column,  
  showPercentages = TRUE,  
  fontSizePercentages = 4  
)
```

**Arguments**

<code>df</code>	A data.frame-like object (such that <code>df[[column]]</code> works).
<code>idx</code>	A numeric vector representing the row indexes of <code>df</code> corresponding to the subset of interest. Can also be a named list of index vectors if multiple subsets are of interest.
<code>column</code>	A character scalar corresponding to a column of <code>df</code> and representing the variable for which the composition should be calculated.
<code>showPercentages</code>	Logical scalar, indicating whether relative frequencies of each category should be shown in the plot.
<code>fontSizePercentages</code>	Numerical scalar, indicating the font size of the relative frequencies, if <code>showPercentages</code> is TRUE.

**Value**

A ggplot object.

**Author(s)**

Charlotte Soneson

**Examples**

```
df <- data.frame(celltype = sample(LETTERS[1:5], 1000, replace = TRUE,
                                prob = c(0.1, 0.2, 0.5, 0.05, 0.15)))
idx <- sample(seq_len(1000), 200)
compareCompositionPlot(df, idx, "celltype")
```

---

geosketch

*Run geosketch to subsample a matrix*


---

**Description**

Perform geometric sketching with the geosketch python package.

**Usage**

```
geosketch(
  mat,
  N,
  replace = FALSE,
  k = "auto",
  alpha = 0.1,
  seed = NULL,
  max_iter = 200,
  one_indexed = TRUE,
  verbose = FALSE
)
```

**Arguments**

mat	m x n matrix. Samples (the dimension along which to subsample) should be in the rows, features in the columns.
N	Numeric scalar, the number of samples to retain.
replace	Logical scalar, whether to sample with replacement.
k	Numeric scalar or "auto", specifying the number of covering. If k = "auto" (the default), it is set to $\sqrt{\text{nrow}(\text{mat})}$ for <code>replace = TRUE</code> and to N for <code>replace = FALSE</code> .
alpha	Numeric scalar defining the acceptable interval around k. Binary search halts when it obtains between $k * (1 - \alpha)$ and $k * (1 + \alpha)$ covering boxes.
seed	Numeric scalar or NULL (default). If not NULL, it will be converted to integer and passed to numpy to seed the random number generator.
max_iter	Numeric scalar giving the maximum iterations at which to terminate binary search in rare cases of non-monotonicity of covering boxes.
one_indexed	Logical scalar, whether to return one-indexed indices.
verbose	Logical scalar, whether to print logging output while running.

**Details**

The first time this function is run, it will create a conda environment containing the geosketch package. This is done via the basilisk R/Bioconductor package - see the documentation for that package for troubleshooting.

**Value**

A numeric vector with indices to retain.

**Author(s)**

Charlotte Soneson, Michael Stadler

**References**

Hie et al (2019): Geometric sketching compactly summarizes the single-cell transcriptomic landscape. Cell Systems 8, 483–493.

**Examples**

```
x <- matrix(rnorm(500), nrow = 100)
geosketch(mat = x, N = 10, seed = 42)
```

---

getGeosketchNames	<i>Get names of geosketch functions</i>
-------------------	---

---

**Description**

Get names of geosketch functions

**Usage**

```
getGeosketchNames()
```

**Value**

A list of names of objects exposed in the geosketch module

**Author(s)**

Charlotte Soneson

**Examples**

```
getGeosketchNames()
```

getScSamplerNames      *Get names of scSampler functions*

---

**Description**

Get names of scSampler functions

**Usage**

```
getScSamplerNames()
```

**Value**

A list of names of objects exposed in the scSampler module

**Author(s)**

Charlotte Soneson

**Examples**

```
getScSamplerNames()
```

---

hausdorffDistPlot      *Create diagnostic plot of Hausdorff distances*

---

**Description**

Create diagnostic plot showing the Hausdorff distance between a sketch and the full data set, for varying sketch sizes. For reproducibility, seed the random number generator before calling this function using `set.seed`.

**Usage**

```
hausdorffDistPlot(  
  mat,  
  Nvec,  
  Nrep = 5,  
  q = 1e-04,  
  methods = c("geosketch", "scsampler", "uniform"),  
  extraArgs = list()  
)
```

**Arguments**

mat	m x n matrix. Samples (the dimension along which to subsample) should be in the rows, features in the columns.
Nvec	Numeric vector of sketch sizes.
Nrep	Numeric scalar indicating the number of sketches to draw for each sketch size.
q	Numeric scalar in [0,1], indicating the fraction of largest minimum distances to discard when calculating the robust Hausdorff distance. Setting q=0 gives the classical Hausdorff distance. The default is 1e-4, as suggested by Hie et al (2019).
methods	Character vector, indicating which method(s) to include in the plot. Should be a subset of c("geosketch", "scsampler", "uniform"), where "uniform" randomly samples from input features with uniform probabilities.
extraArgs	Named list providing extra arguments to the respective methods (beyond the matrix and the sketch size). The names of the list should be the method names (currently, "geosketch" or "scsampler"), and each list element should be a named list of argument values. See the examples for an illustration of how to use this argument. Note that the seed argument, if provided to any of the methods, will be ignored (since it would imply providing the same seed for each repeated run of the sketching).

**Value**

A ggplot object.

**Author(s)**

Charlotte Soneson, Michael Stadler

**References**

Hie et al (2019): Geometric sketching compactly summarizes the single-cell transcriptomic landscape. *Cell Systems* 8, 483–493.

Song et al (2022): scSampler: fast diversity-preserving subsampling of large-scale single-cell transcriptomic data. *bioRxiv* doi:10.1101/2022.01.15.476407

Huttenlocher et al (1993): Comparing images using the Hausdorff distance. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 15(9), 850-863.

**Examples**

```
## Generate example data matrix
mat <- matrix(rnorm(1000), nrow = 100)

## Generate diagnostic Hausdorff distance plot
## (including all available methods)
hausdorffDistPlot(mat, Nvec = c(10, 25, 50))

## Provide additional arguments for geosketch
```

```
hausdorfffDistPlot(mat, Nvec = c(10, 25, 50), Nrep = 2,  
  extraArgs = list(geosketch = list(max_iter = 100)))
```

---

scsampler

*Run scSampler to subsample a matrix*

---

## Description

Perform subsampling with the scSampler python package.

## Usage

```
scsampler(mat, N, random_split = 1, seed = 0)
```

## Arguments

mat	m x n matrix. Samples (the dimension along which to subsample) should be in the rows, features in the columns.
N	Numeric scalar, the number of samples to retain.
random_split	Numeric scalar, the number of parts to randomly split the data into before subsampling within each part. A larger value will speed up computations, but give less optimal results.
seed	Numeric scalar, passed to scsampler to seed the random number generator.

## Details

The first time this function is run, it will create a conda environment containing the scSampler package. This is done via the basilisk R/Bioconductor package - see the documentation for that package for troubleshooting.

## Value

A numeric vector with indices to retain.

## Author(s)

Charlotte Soneson, Michael Stadler

## References

Song et al (2022): scSampler: fast diversity-preserving subsampling of large-scale single-cell transcriptomic data. bioRxiv doi:10.1101/2022.01.15.476407

## Examples

```
x <- matrix(rnorm(500), nrow = 100)  
scsampler(mat = x, N = 10)
```



# Index

\* **internal**

sketchR-package, [2](#)

compareCompositionPlot, [3](#)

geosketch, [4](#)

getGeosketchNames, [5](#)

getScSamplerNames, [6](#)

hausdorffDistPlot, [6](#)

scsampler, [8](#)

sketchR (sketchR-package), [2](#)

sketchR-package, [2](#)