Package: GeneScape (via r-universe)

October 31, 2026

Type Package

Title Simulate single cell and spatial transcriptomics data

Version 1.2

Date 2023-08-26

Author Qi Gao

Maintainer Qi Gao <qi.gao@duke.edu>

Description Simulate single cell and spatial transcriptomics data with complicated heterogeneity structure with or without reference dataset.

Encoding UTF-8

License GPL (>= 3)

Imports Rcpp (>= 1.0.10), concaveman, MASS (>= 7.3-53.1), corpcor (>= 1.6.10), stats, sf, lamW

LinkingTo Rcpp, RcppArmadillo

RoxygenNote 6.0.1

Config/pak/sysreqs libabsl-dev cmake libgdal-dev gdal-bin libgeos-dev make libssl-dev libproj-dev libsqlite3-dev libudunits2-dev libnode-dev

Repository https://qigao1.r-universe.dev

Date/Publication 2023-08-26 05:14:10 UTC

RemoteUrl https://github.com/qigao1/genescape

RemoteRef HEAD

RemoteSha be6c631fd163574c860f9ba8df416e647e2c4276

Contents

add_distort 2
aggregate_expression 3
assign_cell_types 4
assign_tissue_types 5
cell_position_simulation 6

changeLibSize 6
cumsum_cpp 7
downSampleRead 8
est_bound 8
estimate_gamma 9
estimate_lognormal 10
estimate_nb 10
estimate_zinb 11
estimate_zip 11
fcsim 12
GeneScape_est 13
GeneScape_par 13
GeneScape_sim 15
GeneScapeS_est 16
GeneScapeS_sim 17
inverse_logit 18
knn_tissue_update 18
knnassign 19
rotate_pg 19
rotate_sample 20
sample_int 21
sample_region_simulation 21
scale_sample 22
shift_sample 23
sp_dist_euclidean_cpp 24
spot_simulate 24
tissue_region_simulation 25
update_sample_region 26
zinb_nll 26

Index **28**

`add_distort` *add_distort*

Description

The function adds distortions to the input tissue type regions.

Usage

```
add_distort(old_tissue_region, sample_angle = pi/6, tissue_angle = 0,
  sigma = 0.01, scaleloc = FALSE, thres = 1)
```

Arguments

old_tissue_region	An sf object of the tissue type region polygons.
sample_angle	a number of the angles for the whole sample.
tissue_angle	a number or a vector of the angles for each of the tissue regions. Tissue type regions are rotated clockwise by the input angle.
sigma	standard deviation of the normal error
scaleloc	whether to scale the location coordinates to [0,1]
thres	threshold parameter of the concave hull calculation method. Larger value results in simpler shapes

Details

The function adds distortions to the input tissue type regions by adding normal errors to the coordinates of the vertices of the tissue type region polygons.

Value

An sf object of the polygons of the updated tissue type regions.

References

Pebesma, E., 2018. Simple Features for R: Standardized Support for Spatial Vector Data. The R Journal 10 (1), 439-446, https://doi.org/10.32614/RJ-2018-009

---

aggregate_expression *aggregate_expression*

Description

The function aggregate the gene expression level in high resolution data to generate low resolution data.

Usage

```
aggregate_expression(counts, cell_position, spot_position,
  dis_thres_prop = 0.03, sigma_prop = 0.01, lib_size_mean = 5000,
  lib_size_dispersion = 3, cell_annotation = NULL)
```

Arguments

counts	read count matrix
cell_position	coordinates of the cells
spot_position	coordinates of the spots
dis_thres_prop	radius of the spot
sigma_prop	standard deviation of normal distribution. larger sigma implies that cells outside of the spot are more likely to contribute to the read observed in the spot
lib_size_mean	mean parameter of target library size (negative binomial distribution)
lib_size_dispersion	dispersion parameter of target library size (negative binomial distribution)
cell_annotation	cell type annotations

Details

This function aggregate the gene expression level in high resolution (cell level) data to generate low resolution (spot level) data. Cells contribute to the spots based on the distance between cell centers and spot centers, with the cells inside the spot contribute all their reads to the correponding spot. After obtaining the read pool, the funtion also downsample the reads to a target mean spot library size.

Value

A matrix of the gene (row) expression in each spot (column)

---

assign_cell_types *assign_cell_types*

Description

The function chooses the cell type for each cell.

Usage

```
assign_cell_types(ttype, nttype, cell_type_proportion = NULL)
```

Arguments

ttype	tissue types of the cells
nttype	total number of possible tissue types. should be no less than the max value of ttype and the number of unique values in ttype. some tissue types may not exist in ttype
cell_type_proportion	matrix of cell type proportion, cell type (row) by tissue type (column)

Details

This function choose the cell type for each cell. Each tissue type could contains multiple types of cells. Based on the cell type proportion, this function randomly samples the cell type for each cell based on their corresponding tissue type.

Value

A vector of the assigned cell type

`assign_tissue_types` *assign_tissue_types*

Description

The function assigns to each cell the corresponding tissue type.

Usage

```
assign_tissue_types(tissue_region, cell_spot_position, nneighbor = 3)
```

Arguments

`tissue_region`	sf object of tissue region polygons.
`cell_spot_position`	coordinates for the cells
`nneighbor`	number of nearest tissue region taken into account

Details

This function assigns the cells to a tissue type if the cell is located in the tissue region polygon

Value

A vector of the tissue region type

References

Pebesma, E., 2018. Simple Features for R: Standardized Support for Spatial Vector Data. The R Journal 10 (1), 439-446, https://doi.org/10.32614/RJ-2018-009

`cell_position_simulation` *cell_position_simulation*

Description

The function simulates the coordinates of the cells within the sample region.

Usage

```
cell_position_simulation(sample_vertices, ncells = 10000)
```

Arguments

`sample_vertices`	coordinates for the vertices of sample polygon. Each row for a vertex.
`ncells`	number of cells to simulate

Details

This function simulates the location of the cells within the sample region

Value

A matrix of the centers, 1st column as x coordinate and 2nd column for the y coordinate

References

Pebesma, E., 2018. Simple Features for R: Standardized Support for Spatial Vector Data. The R Journal 10 (1), 439-446, https://doi.org/10.32614/RJ-2018-009

`changeLibSize` *changeLibSize This function modifies the simulated counts so that the total read count in each spot is the same as the example data.*

Description

changeLibSize This function modifies the simulated counts so that the total read count in each spot is the same as the example data.

Usage

```
changeLibSize(count, nreadspot, weight)
```

Arguments

count	simulated matrix of read count, rows as genes and columns as spots
nreadspot	number of total read count in each spot in the example data
weight	a matrix of probability weights for the read count of each gene in each cell being changed.

Value

a matrix of the updated read count

Author(s)

Qi Gao

---

cumsum_cpp *cumsum_cpp This function calculates the cumulative sum (partial sum) of a vector.*

Description

cumsum_cpp This function calculates the cumulative sum (partial sum) of a vector.

Usage

```
cumsum_cpp(x)
```

Arguments

x	vector to be summed.

Value

vector of the cumulative sum (partial sum)

Author(s)

Qi Gao

`downSampleRead` *downSampleRead This function uses down-sampling to obtain the reads in each spot.*

Description

downSampleRead This function uses down-sampling to obtain the reads in each spot.

Usage

```
downSampleRead(count, nread)
```

Arguments

`count`	gene (row) by cell (column) matrix. Gene read count in each cell.
`nread`	cell (row) by spot (column) matrix. Target number of reads sampled from each cell in each spot.

Value

gene (row) by spot (column) matrix. Gene expression level in each spot.

Author(s)

Qi Gao

`est_bound` *est_bound*

Description

The function estimates tissue type regions using cell/spot locations and the corresponding tissue type.

Usage

```
est_bound(sploc, ttype, nttype, thres = 1)
```

Arguments

`sploc`	coordinates for the cells.
`ttype`	tissue types of the cells. should be integers starting from 1.
`nttype`	total number of possible tissue types. should be no less than the max value of ttype and the number of unique values in ttype. some tissue types may not exist in ttype
`thres`	threshold parameter of the concave hull calculation method. Larger value results in simpler shapes

Details

The function estimates tissue type regions using cell/spot locations and the corresponding tissue type from reference data.

Value

A sf object of the polygons of the tissue type regions estimated from data.

References

Joel Gombin and Ramnath Vaidyanathan and Vladimir Agafonkin (2020). concaveman: A Very Fast 2D Concave Hull Algorithm. R package version 1.1.0. https://CRAN.R-project.org/package=concaveman

Pebesma, E., 2018. Simple Features for R: Standardized Support for Spatial Vector Data. The R Journal 10 (1), 439-446, https://doi.org/10.32614/RJ-2018-009

`estimate_gamma` *estimate_gamma This function estimates gamma distribution shape and rate parameters.*

Description

estimate_gamma This function estimates gamma distribution shape and rate parameters.

Usage

```
estimate_gamma(x)
```

Arguments

`x`	target data vector

Value

list of gamma distribution shape and rate parameters

Author(s)

Qi Gao

`estimate_lognormal` *estimate_lognormal This function estimates estimate_lognormal distribution loc and scale parameters.*

Description

estimate_lognormal This function estimates estimate_lognormal distribution loc and scale parameters.

Usage

```
estimate_lognormal(x)
```

Arguments

`x`	target data vector

Value

list of lognormal distribution loc and scale parameters

Author(s)

Qi Gao

`estimate_nb` *estimate_nb This function estimates negative binomial distribution mean and dispersion parameters.*

Description

estimate_nb This function estimates negative binomial distribution mean and dispersion parameters.

Usage

```
estimate_nb(datavec)
```

Arguments

`datavec`	target data vector

Value

vector of negative binomial distribution mean (1st element), dispersion (2nd element) parameters and 0 (3rd element).

Author(s)

Qi Gao

`estimate_zinb` *estimate_zinb This function estimates the mean, dispersion and zero inflation parameters in zero-inflated negative binomial distribution.*

Description

estimate_zinb This function estimates the mean, dispersion and zero inflation parameters in zero-inflated negative binomial distribution.

Usage

```
estimate_zinb(data, maxiter = 400)
```

Arguments

`data`	target data matrix, rows as genes and columns as spots
`maxiter`	maximum number of iteration for R optim

Value

matrix of negative binomial distribution mean (1st column), dispersion (2nd column) and zero-inflation (3rd column) parameters.

Author(s)

Qi Gao

`estimate_zip` *estimate_zip*

Description

This function estimate zero-inflated poisson distribution parameters from data vector

Usage

```
estimate_zip(datavec)
```

Arguments

`datavec`	target data vector

Value

a vector of zero-inflated poisson distribution parameters. first element is mean parameter, while second element is zero-inflation parameter.

References

S. Dencks, M. Piepenbrock and G. Schmitz, "Assessing Vessel Reconstruction in Ultrasound Localization Microscopy by Maximum Likelihood Estimation of a Zero-Inflated Poisson Model," in IEEE Transactions on Ultrasonics, Ferroelectrics, and Frequency Control, vol. 67, no. 8, pp. 1603-1612, Aug. 2020, doi: 10.1109/TUFFC.2020.2980063.

`fcsim` *fcsim*

Description

This function similate differential expression fold change level

Usage

```
fcsim(n.gene, de.id, fc.mean, fc.sd)
```

Arguments

`n.gene`	total number of genes
`de.id`	index of differentially expressed genes
`fc.mean`	location parameter for fold change (normal distribution)
`fc.sd`	scale parameter for fold change (normal distribution)

References

Zappia, L., Phipson, B., & Oshlack, A. (2017). Splatter: Simulation of single-cell RNA sequencing data. Genome Biology, 18(1). https://doi.org/10.1186/s13059-017-1305-0

GeneScape_est *GeneScape_est*

Description

This function estimate zero-inflated poisson distribution parameters from data

Usage

```
GeneScape_est(data, groups = NULL)
```

Arguments

data	target gene (row) by cell (column) data matrix
groups	group of the cells

Value

list of two gene (row) by group (column) matrices. first matrix contains the mean parameter, while second matrix contains zero inflation parameters.

GeneScape_par *GeneScape_par*

Description

This function simulate parameters for single cell RNAseq data simulation with complicated differential expression.

Usage

```
GeneScape_par(nCells = 6000, nGroups = NULL, groups = NULL,
  lib.size = NULL, lib.size.mean = 10000, lib.size.sd = 2000,
  de.fc.mat = NULL, nGenes = 5000, gene.mean.shape = 0.3,
  gene.mean.rate = 0.15, gene.means = NULL, zero.inflat = NULL,
  add.zero.inflat = TRUE, zero.inflat.times = 0.5, de.n = 50,
  de.share = NULL, de.id = NULL, de.fc.mean = 1, de.fc.sd = 0.2,
  add.sub = FALSE, sub.major = NULL, sub.prop = 0.1, sub.group = NULL,
  sub.de.n = 20, sub.de.id = NULL, sub.de.common = FALSE,
  sub.de.fc.mean = 1, sub.de.fc.sd = 0.2)
```

Arguments

nCells	number of cells
nGroups	number of cell groups
groups	group information for cells
lib.size	library size for cells
lib.size.mean	location parameter for library size (log-normal distribution)
lib.size.sd	scale parameter for library size (log-normal distribution)
de.fc.mat	differential expression fold change matrix, could be generated by this function
nGenes	number of genes
gene.mean.shape	shape parameter for mean expression level (Gamma distribution)
gene.mean.rate	rate parameter for mean expression level (Gamma distribution)
gene.means	mean gene expression levels
zero.inflat	vector of zero inflation paramters
add.zero.inflat	whether to add zero inflation
zero.inflat.times	the zero inflation paramter is simulated as zero.inflat.times / (1 + exp(gene.means))
de.n	number of differentially expressed genes in each cell type. Should be a integer or a vector of length nGroups
de.share	number of shared DE genes between neighbor cell types. Should be a vector of length (nGroups - 1)
de.id	the index of genes that are DE across cell types. Should be a list of vectors. Each vector corresponds to a cell type. With non-null value of de.id, de.n and de.share would be ignored.
de.fc.mean	the location parameter for the fold change of DE genes. Should be a number, a vector of length nGroups
de.fc.sd	the scale parameter for fold change (log-normal distribution). Should be a number or a vector of length nGroups
add.sub	whether to add sub-cell-types
sub.major	the major cell types correspond to the sub-cell-types
sub.prop	proportion of sub-cell-types in the corresponding major cell type
sub.group	cell index for sub-cell-types. With non-null sub.group specified, sub.prop would be ignored.
sub.de.n	number of differentially expressed genes in each sub-cell-type compared to the corresponding major cell type. Should be a integer or a vector of length sub.major
sub.de.id	the index of additional differentially expressed genes between sub-cell-types and the corresponding major cell types
sub.de.common	whether the additional differential expression structure should be same for all sub-cell-types

sub.de.fc.mean	similar to de.fc.mean, but for addtional differentially expressed genes in sub-cell-types
sub.de.fc.sd	similar to de.fc.sd, but for addtional differentially expressed genes in sub-cell-types

Value

A list of simulation parameters: vector of cell groups, vector of cell library size, gene expression parameter (zero-inflated poisson model), and matrix of differentially expression fold changes.

References

Zappia, L., Phipson, B., & Oshlack, A. (2017). Splatter: Simulation of single-cell RNA sequencing data. Genome Biology, 18(1). https://doi.org/10.1186/s13059-017-1305-0

Examples

```
set.seed(1)
para <- GeneScape_par()
```

GeneScape_sim *GeneScape_sim*

Description

This function generate single cell RNAseq data using simulation parameter.

Usage

```
GeneScape_sim(para, add.path = FALSE, path.n = 4, path.size = 20,
  path.cor = 0.7, path.id = NULL, band.width = 10, add.hub = FALSE,
  hub.n = 10, hub.size = 20, hub.cor = 0.4, hub.id = NULL,
  hub.fix = NULL)
```

Arguments

para	simulation parameter estimated by GeneScape_est or simulated by GeneScape_par
add.path	whether to add pathways (correlated genes)
path.n	number of pathways included. Should be a integer.
path.size	number of correlated genes (length of pathway). Should be a number or a vector of length path.n
path.cor	correlation parameters
path.id	gene index of correlated (pathway) genes. Should be a list of vectors, with each vector represents a pathway. With non-null value of path.id, path.n would be ignored.

band.width	No correlation exists if distance of 2 genes are further than band_width in a pathway
add.hub	whether to add hub genes
hub.n	number of hub genes included. Should be a integer.
hub.size	number of genes correlated to the hub gene. Should be a number or a vector of length hub.n
hub.cor	correlation parameters between hub genes and their correlated genes
hub.id	gene index of hub genes. Should be a list of vectors. With non-null value of hub.id, hub.n would be ignored.
hub.fix	user defined genes correlated to hub genes (others are randomly selected). Should be a list of vectors of length hub.n or same as hub.id.

Value

A list of read count data, cell groups, cell library size, gene mean expression, gene differential expression rate, pathway genes and hub gene indices.

References

Zappia, L., Phipson, B., & Oshlack, A. (2017). Splatter: Simulation of single-cell RNA sequencing data. Genome Biology, 18(1). https://doi.org/10.1186/s13059-017-1305-0

Examples

```
set.seed(1)
para <- GeneScape_par()
data <- GeneScape_sim(para)
```

GeneScapeS_est *GeneScapeS_est This function estimates distribution parameters from an example data.*

Description

GeneScapeS_est This function estimates distribution parameters from an example data.

Usage

```
GeneScapeS_est(data, ttype)
```

Arguments

data	Example data matrix
ttype	Original tissue type of each spot

Value

list of matrices of negative binomial distribution mean (1st column), dispersion (2nd column) and zero-inflation (3rd column) parameters. one matrix for each tissue type.

Author(s)

Qi Gao

`GeneScapeS_sim` *GeneScapeS_sim This function generates simulated data based on distribution parameters estimated from an example data.*

Description

GeneScapeS_sim This function generates simulated data based on distribution parameters estimated from an example data.

Usage

```
GeneScapeS_sim(para, newttype, rank = TRUE, data = NULL, ttype = NULL)
```

Arguments

`para`	Example data matrix
`newttype`	output tissue type of each spot
`rank`	whether to rank the expression level in each tissue type based on the example
`data`	original reference dataset used for create para
`ttype`	Original tissue type of each spot

Value

matrix of simulated data

Author(s)

Qi Gao

`inverse_logit` *inverse_logit This function is the inverse logit function.*

Description

inverse_logit This function is the inverse logit function.

Usage

```
inverse_logit(x)
```

Arguments

x	number

Value

inverse logit function value with input x

Author(s)

Qi Gao

`knn_tissue_update` *knn_tissue_update*

Description

The function updates the tissue type for each cell.

Usage

```
knn_tissue_update(cell_spot_position, old_tissue_type, k)
```

Arguments

cell_spot_position	coordinates for the cells
old_tissue_type	previously assigned tissue type for each cell. should be integers start from one
k	number of nearest neighbor

Details

This function assigns to the each cell to a new tissue type by sampling the tissue types of its k nearest neighbors.

Value

A vector of the updated tissue type

`knnassign` *knnassign This function uses KNN sampling to simulate gradually changed tissue types.*

Description

knnassign This function uses KNN sampling to simulate gradually changed tissue types.

Usage

```
knnassign(distmat, ttype, nttype, k)
```

Arguments

`distmat`	Spot distance matrix
`ttype`	Input tissue type of each spot
`nttype`	total number of possible tissue types. should be no less than the max value of ttype and the number of unique values in ttype. some tissue types may not exist in ttype
`k`	parameter for KNN sampling

Value

List of tissue type weight and updated tissue type with gradual change

Author(s)

Qi Gao

`rotate_pg` *rotate_pg*

Description

The function rotate polygons clockwise.

Usage

```
rotate_pg(pg, angle = pi/12, center = NULL)
```

Arguments

pg	An sf object of the polygon.
angle	a number of the angle. The polygon is rotated clockwise by the input angle.
center	customized rotation center

Details

The function rotate polygons clockwise based on the input angle around the center (by default its centroid).

Value

An sf object of the rotated polygon.

References

Pebesma, E., 2018. Simple Features for R: Standardized Support for Spatial Vector Data. The R Journal 10 (1), 439-446, https://doi.org/10.32614/RJ-2018-009

---

rotate_sample *rotate_sample*

Description

The function rotate polygons clockwise.

Usage

```
rotate_sample(tissue_region, angle = pi/12, thres = 1)
```

Arguments

tissue_region	Tissue region polygons.
angle	a number of the angle. The polygon is rotated clockwise by the input angle.
thres	threshold parameter of the concave hull calculation method. Larger value results in simpler shapes

Details

The function rotate tissue regions clockwise based on the input angle around the center (by default its centroid).

Value

An sf object of the rotated tissue regions

References

Pebesma, E., 2018. Simple Features for R: Standardized Support for Spatial Vector Data. The R Journal 10 (1), 439-446, https://doi.org/10.32614/RJ-2018-009

`sample_int` *sample_int This function samples integers from 1 to an upper limit integer value with equal probability without replacement.*

Description

sample_int This function samples integers from 1 to an upper limit integer value with equal probability without replacement.

Usage

```
sample_int(maxvalue, nsample)
```

Arguments

`maxvalue`	the max integer to be sampled.
`nsample`	the number of integer to be sampled.

Value

vector of the integers in the sample

Author(s)

Qi Gao

`sample_region_simulation` *sample_region_simulation*

Description

This function simulates the sample region.

Usage

```
sample_region_simulation(nvert = 13, sample_shape = c("square", "circle"),
  limitx = c(0, 1), limity = c(0, 1))
```

Arguments

nvert	number of vertices of the sample polygon
sample_shape	shape of the capture area
limitx	min and max value of x coordinates
limity	min and max value of y coordinates

Details

This function simulates the sample region with the capture area as a polygon.

Value

A matrix of the coordinates of sample polygon vertices

References

Valtr, P. Probability that n random points are in convex position. Discrete Comput Geom 13, 637-643 (1995). https://doi.org/10.1007/BF02574070

---

scale_sample *scale_sample*

Description

The function shifts the tissue type regions.

Usage

```
scale_sample(tissue_region, scale_prop = c(1, 1), thres = 1)
```

Arguments

tissue_region	An sf object of the tissue type region polygons.
scale_prop	a vector of length 2, for x and y correspondingly. Value greater than 1 would enlarge the tissue type region on the corresponding coordinates.
thres	threshold parameter of the concave hull calculation method. Larger value results in simpler shapes

Details

The function shifts tissue type regions by modifying the x and y coordinates of the vertices of the tissue type region polygons. The sizes of modifications equal to the input proportion times the difference between the maximum of x (y) coordinates and the minimum of x (y) coordinates

Value

An sf object of the polygons of the updated tissue type regions.

References

Pebesma, E., 2018. Simple Features for R: Standardized Support for Spatial Vector Data. The R Journal 10 (1), 439-446, https://doi.org/10.32614/RJ-2018-009

`shift_sample` *shift_sample*

Description

The function scales the tissue type regions.

Usage

```
shift_sample(tissue_region, shift_prop = c(0, 0))
```

Arguments

`tissue_region`	An sf object of the tissue type region polygons.
`shift_prop`	a vector of length 2, for x and y correspondingly. Large absolute value implies greater shift. Positive value implies positive change of x and y coordinates (shift up and right)

Details

The function scales the tissue type regions around the sample centroid

Value

An sf object of the polygons of the updated tissue type regions.

References

Pebesma, E., 2018. Simple Features for R: Standardized Support for Spatial Vector Data. The R Journal 10 (1), 439-446, https://doi.org/10.32614/RJ-2018-009

`sp_dist_euclidean_cpp` *sp_dist_euclidean_cpp This function calculates Euclidean distance between cell locations and spot locations.*

Description

sp_dist_euclidean_cpp This function calculates Euclidean distance between cell locations and spot locations.

Usage

```
sp_dist_euclidean_cpp(cloc, sloc)
```

Arguments

`cloc`	matrix of location coordinates of the cells. Each row represents a cell.
`sloc`	matrix of location coordinates of the spots. Each row represents a spot.

Value

matrix of the Euclidean distance, number of cells (row) by number of spots (column)

Author(s)

Qi Gao

`spot_simulate` *spot_simulate*

Description

The function simulates the spot positions.

Usage

```
spot_simulate(dimension = c(78, 64), n = 5000, structure = c("interlaced",
  "uniform", "random"), error_prop = 0.04, limitx = c(0, 1), limity = c(0,
  1))
```

Arguments

`dimension`	dimension of the spots, only used for "interlaced" and "uniform" structure.
`n`	number of spots, only used for "random" structure
`structure`	structure of the spots
`error_prop`	size of error added to the coordinates of the spots
`limitx`	min and max value of x coordinates
`limity`	min and max value of y coordinates

Details

This function simulates the spot coordinates. The "interlaced" structure mimic the Visium sequencing data. The "uniform" structure generate spot positions that are located on the vertices of squares. And the "random" structure randomly sample the spot locations.

Value

A matrix of the spot coordinates

`tissue_region_simulation`
tissue_region_simulation

Description

The function simulates regions of different tissue types within the sample region.

Usage

```
tissue_region_simulation(sample_vertices, ntissuetype = 4, centers = NULL)
```

Arguments

`sample_vertices`	coordinates for the vertices of sample polygon. Each row for a vertex.
`ntissuetype`	number of tissue types
`centers`	centers of the tissue types

Details

This function simulates the tissue type regions using voronoi polygons correponding to the simulated or input centers

Value

A sf object of the polygons of the tissue type regions.

References

Pebesma, E., 2018. Simple Features for R: Standardized Support for Spatial Vector Data. The R Journal 10 (1), 439-446, https://doi.org/10.32614/RJ-2018-009

update_sample_region *update_sample_region*

Description

The function updates the sample region based on the regions of different tissue types.

Usage

```
update_sample_region(tissue_region, thres = 1)
```

Arguments

tissue_region	An sf object of the tissue type region polygons.
thres	threshold parameter of the concave hull calculation method. Larger value results in simpler shapes

Details

The function updates the sample region by finding the concave hull of the vertices of tissue type region polygons.

Value

An sf object of the polygons of the updated tissue type regions.

References

Pebesma, E., 2018. Simple Features for R: Standardized Support for Spatial Vector Data. The R Journal 10 (1), 439-446, https://doi.org/10.32614/RJ-2018-009

zinb_nll *zinb_nll This function calculates the negative log likelihood of zero-inflated negative binomial distribution.*

Description

zinb_nll This function calculates the negative log likelihood of zero-inflated negative binomial distribution.

Usage

```
zinb_nll(y, par)
```

Arguments

y	observations
par	vector of 3 elements. parameters (mean, dispersion, zero-inflation) for zero-inflated negative binomial distribution.

Value

negative log likelihood of zero-inflated negative binomial distribution

Author(s)

Qi Gao

Index

add_distort, 2
aggregate_expression, 3
assign_cell_types, 4
assign_tissue_types, 5

cell_position_simulation, 6
changeLibSize, 6
cumsum_cpp, 7

downSampleRead, 8

est_bound, 8
estimate_gamma, 9
estimate_lognormal, 10
estimate_nb, 10
estimate_zinb, 11
estimate_zip, 11

fcsim, 12

GeneScape_est, 13
GeneScape_par, 13
GeneScape_sim, 15
GeneScapeS_est, 16
GeneScapeS_sim, 17

inverse_logit, 18

knn_tissue_update, 18
knnassign, 19

rotate_pg, 19
rotate_sample, 20

sample_int, 21
sample_region_simulation, 21
scale_sample, 22
shift_sample, 23
sp_dist_euclidean_cpp, 24
spot_simulate, 24

tissue_region_simulation, 25

update_sample_region, 26

zinb_nll, 26