

Package: ProFAST (via r-universe)

October 4, 2026

Type Package

Title Probabilistic Factor Analysis for Spatially-Aware Dimension Reduction

Version 1.9

Date 2026-06-04

Author Wei Liu [aut, cre], Xiao Zhang [aut], Jin Liu [aut]

Maintainer Wei Liu <liuweideng@gmail.com>

Description Probabilistic factor analysis for spatially-aware dimension reduction across multi-section spatial transcriptomics data with millions of spatial locations. More details can be referred to Wei Liu, et al. (2023) <[doi:10.1101/2023.07.11.548486](https://doi.org/10.1101/2023.07.11.548486)>.

License GPL-3

Depends R (>= 4.0.0), gtools

Imports Rcpp (>= 1.0.10), furrr, purrr, future, ggplot2, DR.SC, Matrix, mclust, pbapply, irlba, Seurat, parallel, methods, stats, utils

LazyData true

URL <https://github.com/feiyoun/ProFAST>

BugReports <https://github.com/feiyoun/ProFAST/issues>

Suggests knitr, rmarkdown, harmony, PRECAST, performance, nnet, biomaRt, scater, ggrepel, RANN, grDevices

LinkingTo Rcpp, RcppArmadillo

VignetteBuilder knitr

Encoding UTF-8

RoxygenNote 7.3.2

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

Repository <https://feiyoun.r-universe.dev>

Date/Publication 2026-06-05 09:55:12 UTC
RemoteUrl https://github.com/feiyoungh/profast
RemoteRef HEAD
RemoteSha 5c347782852ede0cbcdf39f53f51caf189db8b28

Contents

AddAdj	2
AddParSettingFAST	4
coembed_plot	4
coembedding_umap	6
CosMx_subset	7
diagnostic.cor.eigs	8
FAST	10
FAST_run	10
FAST_single	12
FAST_structure	13
find.signature.genes	14
get.top.signature.dat	15
get_r2_mcfadden	16
IntegrateSRTData	17
iscmeb_run	18
model_set_FAST	19
NCFM	20
NCFM_fast	21
pbmc3k_subset	22
pdistance	23
RunHarmonyLouvain	24
RuniSCMEB	24
SelectHKgenes	25
top5_signatures	25
transferGeneNames	26
Index	27

AddAdj	<i>Calculate the adjacency matrix given a spatial coordinate matrix</i>
--------	---

Description

Calculate the adjacency matrix given a spatial coordinate matrix with 2-dimension or 3-dimension or more.

Usage

```
AddAdj(
  pos,
  type = "fixed_distance",
  platform = c("Others", "Visium", "ST"),
  neighbors = 6,
  ...
)
```

Arguments

<code>pos</code>	a matrix object, with columns representing the spatial coordinates that can be any diemson, i.e., 2, 3 and >3.
<code>type</code>	an optional string, specify which type of neighbors' definition. Here we provide two definition: one is "fixed_distance", the other is "fixed_number".
<code>platform</code>	a string, specify the platform of the provided data, default as "Others". There are more platforms to be chosen, including "Visium", "ST" and "Others" ("Others" represents the other SRT platforms except for 'Visium' and 'ST') The platform helps to calculate the adjacency matrix by defining the neighborhoods when type="fixed_distance" is chosen.
<code>neighbors</code>	an optional postive integer, specify how many neighbors used in calculation, default as 6.
<code>...</code>	Other arguments passed to getAdj_auto .

Details

When the type = "fixed_distance", then the spots within the Euclidean distance cutoffs from one spot are regarded as the neighbors of this spot. When the type = "fixed_number", the K-nearest spots are regarded as the neighbors of each spot.

Value

return a sparse matrix, representing the adjacency matrix.

References

None

See Also

None

Examples

```
data(CosMx_subset)
pos <- as.matrix(CosMx_subset@meta.data[,c("x", "y")])
Adj_sp <- AddAdj(pos)
```

AddParSettingFAST	<i>Add FAST model settings for a PRECASTObj object</i>
-------------------	--

Description

Add FAST model settings for a PRECASTObj object

Usage

```
AddParSettingFAST(PRECASTObj, ...)
```

Arguments

PRECASTObj	a PRECASTObj object created by CreatePRECASTObject .
...	other arguments to be passed to model_set_FAST function.

Value

Return a revised PRECASTObj object with slot parameterList changed.

References

None

coembed_plot	<i>Coembedding dimensional reduction plot</i>
--------------	---

Description

Graph output of a dimensional reduction technique on a 2D scatter plot where each point is a cell or feature and it's positioned based on the coembeddings determined by the reduction technique. By default, cells and their signature features are colored by their identity class (can be changed with the group.by parameter).

Usage

```
coembed_plot(
  seu,
  reduction,
  gene_txtdata = NULL,
  cell_label = NULL,
  xy_name = reduction,
  dims = c(1, 2),
  cols = NULL,
  shape_cg = c(1, 5),
```

```

    pt_size = 1,
    pt_text_size = 5,
    base_size = 16,
    base_family = "serif",
    legend.point.size = 5,
    legend.key.size = 1.5,
    alpha = 0.3
  )

```

Arguments

<code>seu</code>	a Seurat object with coembedding in the reductions slot with component name reduction.
<code>reduction</code>	a string, specify the reduction component that denotes coembedding.
<code>gene_txtdata</code>	a data.frame object with columns including 'gene' and 'label', specify the cell type/spatial domain and signature genes. Default as NULL, all features will be used in coembeddings.
<code>cell_label</code>	an optional character in columns of metadata, specify the group of cells/spots. Default as NULL, use Idents as the group.
<code>xy_name</code>	an optional character, specify the names of x and y-axis, default as the same as reduction.
<code>dims</code>	a positive integer vector with length 2, specify the two components for visualization.
<code>cols</code>	an optional string vector, specify the colors for cell group in visualization.
<code>shape_cg</code>	a positive integers with length 2, specify the shapes of cell/spot and feature in plot.
<code>pt_size</code>	an optional integer, specify the point size, default as 1.
<code>pt_text_size</code>	an optional integer, specify the point size of text, default as 5.
<code>base_size</code>	an optional integer, specify the basic size.
<code>base_family</code>	an optional character, specify the font.
<code>legend.point.size</code>	an optional integer, specify the point size of legend.
<code>legend.key.size</code>	an optional integer, specify the size of legend key.
<code>alpha</code>	an optional positive real, range from 0 to 1, specify the transparency of points.

Details

None

Value

return a ggplot object

References

None

See Also

[coembedding_umap](#)

Examples

```
data(pbmc3k_subset)
data(top5_signatures)
coembed_plot(pbmc3k_subset, reduction = "UMAPsig",
  gene_txtdata = top5_signatures, pt_text_size = 3, alpha=0.3)
```

coembedding_umap

Calculate UMAP projections for coembedding of cells and features

Description

Calculate UMAP projections for coembedding of cells and features

Usage

```
coembedding_umap(
  seu,
  reduction,
  reduction.name,
  gene.set = NULL,
  slot = "data",
  assay = "RNA",
  seed = 1
)
```

Arguments

<code>seu</code>	a Seurat object with coembedding in the reductions slot with component name <code>reduction</code> .
<code>reduction</code>	a string, specify the reduction component that denotes coembedding.
<code>reduction.name</code>	a string, specify the reduction name for the obtained UMAP projection.
<code>gene.set</code>	a string vector, specify the features (genes) in calculating the UMAP projection, default as all features.
<code>slot</code>	an optional string, specify the slot in the assay, default as 'data'.
<code>assay</code>	an optional string, specify the assay name in the Seurat object when adding the UMAP projection.
<code>seed</code>	an optional integer, specify the random seed for reproducibility.

Details

None

Value

return a revised Seurat object by adding a new reduction component named 'reduction.name'.

References

None

See Also

None

Examples

```
data(pbmc3k_subset)
data(top5_signatures)

pbmc3k_subset <- coembedding_umap(
  pbmc3k_subset, reduction = "ncfm", reduction.name = "UMAPsig",
  gene.set = top5_signatures$gene
)
```

CosMx_subset	<i>A Seurat object including spatial transcriptomics dataset from CosMx platform</i>
--------------	--

Description

This data is a subset of SCLC CosMx spatial transcriptomics dataset.

Usage

```
data(CosMx_subset)
```

Format

A Seurat object, including count matrix, spital coordinates, and manual annotation.

Source

The data is from the CosMx SRT sequencing platform.

References

None

Examples

```
# Show some examples of how to use the dataset.
data(CosMx_subset)
library(Seurat)
CosMx_subset
```

diagnostic.cor.eigs	<i>Determine the dimension of low dimensional embedding</i>
---------------------	---

Description

This function estimate the dimension of low dimensional embedding for a given cell by gene expression matrix. For more details, see Franklin et al. (1995) and Crawford et al. (2010).

Usage

```
diagnostic.cor.eigs(object, ...)
```

```
## Default S3 method:
diagnostic.cor.eigs(
  object,
  q_max = 50,
  plot = TRUE,
  n.sims = 10,
  parallel = TRUE,
  ncores = 10,
  seed = 1,
  ...
)
```

```
## S3 method for class 'Seurat'
diagnostic.cor.eigs(
  object,
  assay = NULL,
  slot = "data",
  nfeatures = 2000,
  q_max = 50,
  seed = 1,
  ...
)
```

Arguments

<code>object</code>	A Seurat or matrix object
<code>...</code>	Other arguments passed to <code>diagnostic.cor.eigs.default</code> .
<code>q_max</code>	the upper bound of low dimensional embedding. Default is 50.
<code>plot</code>	a indicator of whether plot eigen values.
<code>n.sims</code>	number of simulaton times. Default is 10.
<code>parallel</code>	a indicator of whether use parallel analysis.
<code>ncores</code>	the number of cores used in parallel analysis. Default is 10.
<code>seed</code>	a postive integer, specify the random seed for reproducibility
<code>assay</code>	an optional string, specify the name of assay in the Seurat object to be used.
<code>slot</code>	an optional string, specify the name of slot.
<code>nfeatures</code>	an optional integer, specify the number of features to select as top variable features. Default is 2000.

Value

A data.frame with attribute 'q_est' and 'plot', which is the estimated dimension of low dimensional embedding. In addition, this data.frame containing the following components:

- `q` - The index of eigen values.
- `eig_value` - The eigen values on observed data.
- `eig_sim` - The mean value of eigen values of `n.sims` simulated data.
- `q_est` - The selected dimension in `attr(obj, 'q_est')`.
- `plot` - The plot saved in `attr(obj, 'plot')`.

References

1. Franklin, S. B., Gibson, D. J., Robertson, P. A., Pohlmann, J. T., & Fralish, J. S. (1995). Parallel analysis: a method for determining significant principal components. *Journal of Vegetation Science*, 6(1), 99-106.
2. Crawford, A. V., Green, S. B., Levy, R., Lo, W. J., Scott, L., Svetina, D., & Thompson, M. S. (2010). Evaluation of parallel analysis methods for determining the number of factors. *Educational and Psychological Measurement*, 70(6), 885-901.

Examples

```
n <- 100
p <- 50
d <- 15
object <- matrix(rnorm(n*d), n, d) %*% matrix(rnorm(d*p), d, p)
diagnostic.cor.eigs(object, n.sims=2)
```

FAST

*Run FAST model for a PRECASTObj object***Description**

Run FAST model for a PRECASTObj object

Usage

```
FAST(PRECASTObj, q = 15, fit.model = c("poisson", "gaussian"))
```

Arguments

PRECASTObj	a PRECASTObj object created by CreatePRECASTObject .
q	an optional integer, specify the number of low-dimensional embeddings to extract in FAST
fit.model	an optional string, specify the version of FAST to be fitted. The Gaussian version models the log-count matrices while the Poisson version models the count matrices; default as poisson.

Value

Return a revised PRECASTObj object with slot PRECASTObj@resList added by a FAST component.

References

None

FAST_run

*(Variational) ICM-EM algorithm for implementing FAST model***Description**

(Variational) ICM-EM algorithm for implementing FAST model

Usage

```
FAST_run(
  XList,
  AdjList,
  q = 15,
  fit.model = c("gaussian", "poisson"),
  AList = NULL,
  maxIter = 25,
```

```

    epsLogLik = 1e-05,
    verbose = TRUE,
    seed = 1,
    error_heter = TRUE,
    Psi_diag = FALSE,
    Vint_zero = FALSE
  )

```

Arguments

<code>XList</code>	an M-length list consisting of multiple matrices with class <code>dgCMatrx</code> or <code>matrix</code> that specifies the count/log-count gene expression matrix for each data batch used for FAST model.
<code>AdjList</code>	an M-length list of sparse matrices with class <code>dgCMatrx</code> , specify the adjacency matrix used for intrinsic CAR model in FAST. We provide this interface for those users who would like to define the adjacency matrix by themselves.
<code>q</code>	an optional integer, specify the number of low-dimensional embeddings to extract in FAST. Larger <code>q</code> means more information extracted.
<code>fit.model</code>	an optional string, specify the version of FAST to be fitted. The Gaussian version models the log-count matrices while the Poisson versions models the count matrices; default as <code>gaussian</code> due to faster computation.
<code>AList</code>	an optional list with each component being a vector whose length is equal to the rows of component in <code>XList</code> , specify the normalization factor in FAST. The default is <code>NULL</code> that means the normalization factor equal to 1.
<code>maxIter</code>	the maximum iteration of ICM-EM algorithm. The default is 30.
<code>epsLogLik</code>	an optional positive value, tolerance of relative variation rate of the observed pseudo loglikelihood value, default as <code>'1e-5'</code> .
<code>verbose</code>	a logical value, whether output the information in iteration.
<code>seed</code>	a positive integer, the random seed to be set in initialization.
<code>error_heter</code>	a logical value, whether use the heterogeneous error for FAST model, default as <code>TRUE</code> . If <code>error.heter=FALSE</code> , then the homogeneous error is used.
<code>Psi_diag</code>	a logical value, whether set the conditional covariance matrix of the intrinsic CAR to diagonal, default as <code>FALSE</code> .
<code>Vint_zero</code>	an optional logical value, specify whether the initial value of intrinsic CAR component is set to zero; default as <code>FALSE</code> .

Details

None

Value

return a list including the following components: (1) `hV`: an M-length list consisting of spatial embeddings in FAST; (2) `nu`: the estimated intercept vector; (3) `Psi`: the estimated covariance matrix; (4) `W`: the estimated shared loading matrix; (5) `Lam`: the estimated covariance matrix of error term; (6) `ELBO`: the ELBO value when algorithm convergence; (7) `ELBO_seq`: the ELBO values for all iterations.

References

None

See Also[FAST_structure](#), [FAST](#), [model_set_FAST](#)

FAST_single

*Fit FAST model for single-section SRT data***Description**

Fit FAST model for single-section SRT data.

Usage

```
FAST_single(
  seu,
  Adj_sp,
  q = 15,
  fit.model = c("poisson", "gaussian"),
  slot = "data",
  assay = NULL,
  reduction.name = "fast",
  verbose = TRUE,
  ...
)
```

Arguments

<code>seu</code>	a Seurat object.
<code>Adj_sp</code>	a sparse matrix, specify the adjacency matrix among spots.
<code>q</code>	an optional integer, specify the number of low-dimensional embeddings to extract in FAST. Larger q means more information extracted.
<code>fit.model</code>	an optional string, specify the version of FAST to be fitted. The Gaussian version models the log-count matrices while the Poisson version models the count matrices; default as poisson model.
<code>slot</code>	an optional string, specify the slot in Seurat object as the input of FAST model, default as 'data'.
<code>assay</code>	an optional string, specify the assay in Seurat object, default as 'NULL' that means the default assay in Seurat object.
<code>reduction.name</code>	an optional string, specify the reduction name for the fast embedding, default as 'fast'.
<code>verbose</code>	a logical value, whether output the information in iteration.
<code>...</code>	other arguments passed to FAST_run .

Value

return a list including the parameters set in the arguments.

See Also

[FAST_run](#)

FAST_structure	<i>(Variational) ICM-EM algorithm for implementing FAST model with structurized parameters</i>
----------------	--

Description

(Variational) ICM-EM algorithm for implementing FAST model with structurized parameters

Usage

```
FAST_structure(
  XList,
  AdjList,
  q = 15,
  fit.model = c("poisson", "gaussian"),
  parameterList = NULL
)
```

Arguments

XList	an M-length list consisting of multiple matrices with class dgCMatrix or matrix that specify the count/log-count gene expression matrix for each data batch used for FAST model.
AdjList	an M-length list of sparse matrices with class dgCMatrix, specify the adjacency matrix used for intrinsic CAR model in FAST. We provide this interface for those users who would like to define the adjacency matrix by themselves.
q	an optional integer, specify the number of low-dimensional embeddings to extract in FAST
fit.model	an optional string, specify the version of FAST to be fitted. The Gaussian version models the log-count matrices while the Poisson versions models the count matrices; default as gaussian due to faster computation.
parameterList	an optional list, specify other parameters in FAST model; see model_set_FAST for other paramters. The default is NULL that means the default parameters produced by model_set_FAST is used.

Details

None

Value

return a list including the following components: (1) hV: an M-length list consisting of spatial embeddings in FAST; (2) nu: the estimated intercept vector; (3) Psi: the estimated covariance matrix; (4) W: the estimated shared loading matrix; (5) Lam: the estimated covariance matrix of error term; (6) ELBO: the ELBO value when algorithm convergence; (7) ELBO_seq: the ELBO values for all iterations.

References

None

See Also

[FAST_run](#), [FAST](#), [model_set_FAST](#)

find.signature.genes *Find the signature genes for each group of cell/spots*

Description

Find the signature genes for each group of cell/spots based on coembedding distance and expression ratio.

Usage

```
find.signature.genes(
  seu,
  distce.assay = "distce",
  ident = NULL,
  expr.prop.cutoff = 0.1,
  assay = NULL,
  genes.use = NULL
)
```

Arguments

seu	a Seurat object with coembedding in the reductions slot with component name reduction.
distce.assay	an optional character, specify the assay name that contains distance matrix between cells/spots and features, default as 'distce' (distance of coembeddings).
ident	an optional character in columns of metadata, specify the group of cells/spots. Default as NULL, use Idents as the group.
expr.prop.cutoff	an optional positive real ranging from 0 to 1, specify cutoff of expression proportion of features, default as 0.1.
assay	an optional character, specify the assay in seu, default as NULL, representing the default assay in seu.
genes.use	an optional string vector, specify genes as the signature candidates.

Details

In each data.frame object of the returned value, the row.names are gene names, and these genes are sorted by decreasing order of 'distance'. User can define the signature genes as top n genes in distance and that the 'expr.prop' larger than a cutoff. We set the cutoff as 0.1.

Value

return a list with each component a data.frame object having two columns: 'distance' and 'expr.prop'.

References

None

See Also

None

Examples

```
library(Seurat)
data(pbmc3k_subset)
pbmc3k_subset <- pdistance(pbmc3k_subset, reduction='ncfm')
df_list_rna <- find.signature.genes(pbmc3k_subset)
```

get.top.signature.dat *Obtain the top signature genes and related information*

Description

Obtain the top signature genes and related information.

Usage

```
get.top.signature.dat(df.list, ntop = 5, expr.prop.cutoff = 0.1)
```

Arguments

df.list	a list that is obtained by the function find.signature.genes .
ntop	an optional positive integer, specify the how many top signature genes extracted, default as 5.
expr.prop.cutoff	an optional postive real ranging from 0 to 1, specify cutoff of expression proportion of features, default as 0.1.

Details

Using this function, we obtain the top signature genes and organize them into a data.frame. The 'row.names' are gene names. The colname 'distance' means the distance between gene (i.e., VPRED3) and cells with the specific cell type (i.e., B cell), which is calculated based on the co-embedding of genes and cells in the coembedding space. The distance is smaller, the association between gene and the cell type is stronger. The colname 'expr.prop' represents the expression proportion of the gene (i.e., VPRED3) within the cell type (i.e., B cell). The colname 'label' means the cell types and colname 'gene' denotes the gene name. By the data.frame object, we know 'VPRED3' is the one of the top signature gene of B cell.

Value

return a 'data.frame' object with four columns: 'distance', 'expr.prop', 'label' and 'gene'.

References

None

See Also

None

Examples

```
library(Seurat)
data(pbmc3k_subset)
pbmc3k_subset <- pdistance(pbmc3k_subset, reduction='ncfm')
df_list_rna <- find.signature.genes(pbmc3k_subset)
dat.sig <- get.top.signature.dat(df_list_rna, ntop=5)
head(dat.sig)
```

get_r2_mcfadden

Calculate the the adjusted McFadden's pseudo R-square

Description

Calculate the the adjusted McFadden's pseudo R-square between the embeddings and the labels

Usage

```
get_r2_mcfadden(embeds, y)
```

Arguments

embeds	a n-by-q matrix, specify the embedding matrix.
y	a n-length vector, specify the labels.

Details

None

Value

return the adjusted McFadden's pseudo R-square.

References

McFadden, D. (1987). Regression-based specification tests for the multinomial logit model. *Journal of econometrics*, 34(1-2), 63-82.

IntegrateSRTData	<i>Integrate multiple SRT data into a Seurat object</i>
------------------	---

Description

Integrate multiple SRT data based on the PRECASTObj object by FAST and other model fitting.

Usage

```
IntegrateSRTData(
  PRECASTObj,
  seulist_HK,
  Method = c("iSC-MEB", "HarmonyLouvain"),
  seuList_raw = NULL,
  covariates_use = NULL,
  Tm = NULL,
  subsample_rate = 1,
  verbose = TRUE
)
```

Arguments

PRECASTObj	a PRECASTObj object created by CreatePRECASTObject .
seulist_HK	a list with Seurat object as component including only the housekeeping genes.
Method	a string, specify the method to be used and two methods are supported: iSC-MEB and HarmonyLouvain. The default is iSC-MEB.
seuList_raw	an optional list with Seurat object, the raw data.
covariates_use	a string vector, the colnames in PRECASTObj@seulist[[1]]@meta.data, representing other biological covariates to be considered when removing batch effects. This is achieved by adding additional covariates for biological conditions in the regression, such as case or control. Default as 'NULL', denoting no other covariates to be considered.

<code>Tm</code>	an optional numeric vector with the length equal to <code>PRECASTObj@seuList</code> , the time point information if the data include the temporal information. Default as <code>NULL</code> that means there is no temporal information.
<code>subsample_rate</code>	a real ranging in (0,1], specify the rate of spot drawing for speeding up the computation when the number of spots is very large. Default is 1, meaning using all spots.
<code>verbose</code>	an optional logical value, default as <code>TRUE</code> .

Details

If `seuList_raw` is not equal `NULL` or `PRECASTObj@seuList` is not `NULL`, this function will remove the unwanted variations for all genes in `seuList_raw` object. Otherwise, only the unwanted variation of genes in `PRECASTObj@seuList` will be removed. The former requires a big memory to be run, while the latter not. To speed up the computation when the number of spots is very large, we also provide a subsampling schema controlled by the argument `subsample_rate`. When the total number of spots is larger than 80,000, this function will automatically draws 50,000 spots to calculate the parameters in the spatial linear model for removing unwanted variations.

Value

Return a Seurat object by integrating all SRT data batches into a SRT data, where the column "batch" in the meta.data represents the batch ID, and the column "cluster" represents the clusters. The embeddings are put in `seu@reductions` slot and `Idents(seu)` is set to cluster label. Note that only the normalized expression is valid in the data slot while count is invalid.

iscmeb_run

Fit an iSC-MEB model using specified multi-section embeddings

Description

Integrate multiple SRT data based on the `PRECASTObj` by FAST and iSC-MEB model fitting.

Usage

```
iscmeb_run(
  VList,
  AdjList,
  K,
  beta_grid = seq(0, 5, by = 0.2),
  maxIter = 25,
  epsLogLik = 1e-05,
  verbose = TRUE,
  int.model = "EEE",
  init.start = 1,
  Sigma_equal = FALSE,
  Sigma_diag = TRUE,
  seed = 1
)
```

Arguments

VList	a M-length list of embeddings. The i-th element is a $n_i \times q$ matrix, where n_i is the number of spots of sample i, and q is the number of embeddings. We provide this interface for those users who would like to define the embeddings by themselves.
AdjList	an M-length list of sparse matrices with class <code>dgCMatrix</code> , specify the adjacency matrix used for intrinsic CAR model in FAST. We provide this interface for those users who would like to define the adjacency matrix by themselves.
K	an integer, specify the number of clusters.
beta_grid	an optional vector of positive value, the candidate set of the smoothing parameter to be searched by the grid-search optimization approach, default as a sequence starts from 0, ends with 5, increase by 0.2.
maxIter	the maximum iteration of ICM-EM algorithm. The default is 25.
epsLogLik	a string, the species, one of 'Human' and 'Mouse'.
verbose	an optional integer, specify the number of housekeeping genes to be selected.
int.model	an optional string, specify which Gaussian mixture model is used in evaluating the initial values for iSC-MEB, default as "EEE"; and see Mclust for more models' names.
init.start	an optional number of times to calculate the initial value (1 by default). When init.start is larger than 1, initial value will be determined by log likelihood of mclust results.
Sigma_equal	an optional logical value, specify whether Sigmaks are equal, default as FALSE.
Sigma_diag	an optional logical value, specify whether Sigmaks are diagonal matrices, default as TRUE.
seed	an optional integer, the random seed in fitting iSC-MEB model.

Value

returns a `iSCMEBResObj` object which contains all model results.

model_set_FAST	<i>Set parameters for FAST model</i>
----------------	--------------------------------------

Description

Prepare parameters setup for FAST model fitting.

Usage

```
model_set_FAST(
  maxIter = 30,
  epsLogLik = 1e-05,
  error_heter = TRUE,
  Psi_diag = FALSE,
  verbose = TRUE,
  seed = 1
)
```

Arguments

<code>maxIter</code>	the maximum iteration of ICM-EM algorithm. The default is 30.
<code>epsLogLik</code>	an optional positive vlaue, tolerance of relative variation rate of the observed pseudo loglikelihood value, default as '1e-5'.
<code>error_heter</code>	a logical value, whether use the heterogenous error for FAST model, default as TRUE. If <code>error.heter=FALSE</code> , then the homogenous error is used.
<code>Psi_diag</code>	a logical value, whether set the conditional covariance matrices of intrinsic CAR to diagonal, default as FALSE
<code>verbose</code>	a logical value, whether output the information in iteration.
<code>seed</code>	a postive integer, the random seed to be set in initialization.

Value

return a Seurat object with new reduction (named `reduction.name`) added to the 'reductions' slot.

Examples

```
model_set_FAST(maxIter = 30, epsLogLik = 1e-5,
  error_heter=TRUE, Psi_diag=FALSE, verbose=TRUE, seed=2023)
```

NCFM

Cell-feature coembedding for scRNA-seq data

Description

Cell-feature coembedding for scRNA-seq data based on FAST model.

Usage

```
NCFM(
  object,
  assay = NULL,
  slot = "data",
  nfeatures = 2000,
```

```

    q = 10,
    reduction.name = "ncfm",
    weighted = FALSE,
    var.features = NULL
  )

```

Arguments

<code>object</code>	a Seurat object.
<code>assay</code>	an optional string, specify the name of assay in the Seurat object to be used, 'NULL' means default assay in seu.
<code>slot</code>	an optional string, specify the name of slot.
<code>nfeatures</code>	an optional integer, specify the number of features to select as top variable features. Default is 2000.
<code>q</code>	an optional positive integer, specify the dimension of low dimensional embeddings to compute and store. Default is 10.
<code>reduction.name</code>	an optional string, specify the dimensional reduction name, 'ncfm' by default.
<code>weighted</code>	an optional logical value, specify whether use weighted method.
<code>var.features</code>	an optional string vector, specify the variable features used to calculate cell embedding.

Examples

```

data(pbmc3k_subset)
pbmc3k_subset <- NCFM(pbmc3k_subset)

```

NCFM_fast	<i>Cell-feature coembedding for SRT data</i>
-----------	--

Description

Run cell-feature coembedding for SRT data based on FAST model.

Usage

```

NCFM_fast(
  object,
  Adj_sp,
  assay = NULL,
  slot = "data",
  nfeatures = 2000,
  q = 10,
  reduction.name = "fast",
  var.features = NULL,
  ...
)

```

Arguments

<code>object</code>	a Seurat object.
<code>Adj_sp</code>	a sparse matrix, specify the adjacency matrix among spots.
<code>assay</code>	an optional string, the name of assay used.
<code>slot</code>	an optional string, the name of slot used.
<code>nfeatures</code>	an optional positive integer, the number of features to select as top variable features. Default is 2000.
<code>q</code>	an optional positive integer, specify the dimension of low dimensional embeddings to compute and store. Default is 10.
<code>reduction.name</code>	an optional string, dimensional reduction name, 'fast' by default.
<code>var.features</code>	an optional string vector, specify the variable features, used to calculate cell embedding.
<code>...</code>	Other argument passed to the FAST_run .

Examples

```
data(CosMx_subset)
pos <- as.matrix(CosMx_subset@meta.data[,c("x", "y")])
Adj_sp <- AddAdj(pos)
# Here, we set maxIter = 3 for fast computation and demonstration.
CosMx_subset <- NCFM_fast(CosMx_subset, Adj_sp = Adj_sp, maxIter=3)
```

pbmc3k_subset

A Seurat object including scRNA-seq PBMC dataset

Description

This data is a subset of PBMC3k scRNA-seq data in SeuratData package.

Usage

```
data(pbmc3k_subset)
```

Format

A Seurat object, including count matrix, and manual annotation.

Source

The data is from the scRNA-seq sequencing platform.

References

None

Examples

```
# Show examples of how to use the dataset.
data(pbmc3k_subset)
library(Seurat)
pbmc3k_subset
```

pdistance	<i>Calculate the cell-feature distance matrix</i>
-----------	---

Description

Calculate the cell-feature distance matrix based on coembeddings.

Usage

```
pdistance(object, reduction = "fast", assay.name = "distce", eta = 1e-10)
```

Arguments

object	a Seurat object.
reduction	a optional string, dimensional reduction name, 'fast' by default.
assay.name	a optional string, specify the new generated assay name, 'distce' by default.
eta	an optional postive real, a quantity to avoid numerical errors. 1e-10 by default.

Details

This function calculate the distance matrix between cells/spots and features, and then put the distance matrix in a new generated assay. This distance matrix will be used in the siganture gene identification.

Examples

```
data(pbmc3k_subset)
pbmc3k_subset <- NCFM(pbmc3k_subset)
pbmc3k_subset <- pdistance(pbmc3k_subset, "ncfm")
```

RunHarmonyLouvain	<i>Embedding alignment and clustering based on the embeddings from FAST</i>
-------------------	---

Description

Embedding alignment and clustering using the Harmony and Louvain based on the embeddings from FAST as well as determining the number of clusters.

Usage

```
RunHarmonyLouvain(PRECASTObj, resolution = 0.5)
```

Arguments

PRECASTObj	a PRECASTObj object created by CreatePRECASTObject .
resolution	an optional real, the value of the resolution parameter, use a value above (below) 1.0 if you want to obtain a larger (smaller) number of communities.

Value

Return a revised PRECASTObj object with slot PRECASTObj@resList added by a Harmony component (including the aligned embeddings and embeddings of batch effects) and a Louvain component (including the clusters).

Note

This function requires the 'harmony' package for batch correction. If not installed, please run: `install.packages('harmony')` or `remotes::install_github('immunogenomics/harmony')`

RuniSCMEB	<i>Fit an iSC-MEB model using the embeddings from FAST</i>
-----------	--

Description

Fit an iSC-MEB model using the embeddings from FAST and the number of clusters obtained by Louvain.

Usage

```
RuniSCMEB(PRECASTObj, ...)
```

Arguments

PRECASTObj	a PRECASTObj object created by CreatePRECASTObject .
...	other arguments passed to iscmeb_run .

Value

Return a revised PRECASTObj object with an added component iSCMEB in the slot PRECASTObj@resList (including the aligned embeddings, clusters and posterior probability matrix of clusters).

SelectHKgenes	<i>Select housekeeping genes</i>
---------------	----------------------------------

Description

Select housekeeping genes for preparation of removing unwanted variations in expression matrices

Usage

```
SelectHKgenes(seuList, species = c("Human", "Mouse"), HK.number = 200)
```

Arguments

- | | |
|-----------|---|
| seuList | an M-length list consisting of Seurat object, include the information of expression matrix and spatial coordinates (named row and col) in the slot meta.data. |
| species | a string, the species, one of 'Human' and 'Mouse'. |
| HK.number | an optional integer, specify the number of housekeeping genes to be selected. |

Value

Return a string vector of the selected gene names.

top5_signatures	<i>A data.frame object including top five signature genes in scRNA-seq PBMC dataset</i>
-----------------	---

Description

This data is a data.frame object that includes top five signature genes in scRNA-seq PBMC dataset

Usage

```
data(top5_signatures)
```

Format

A data.frame object, including signature genes, distance, and manual annotation.

Source

None

References

None

Examples

```
# Show examples of how to use the dataset.
data(top5_signatures)
head(top5_signatures)
```

transferGeneNames	<i>Transfer gene names from one format to the other format</i>
-------------------	--

Description

Transfer gene names from one format to the other format for two species: human and mouse.

Usage

```
transferGeneNames(
  genelist,
  now_name = "ensembl",
  to_name = "symbol",
  species = c("Human", "Mouse")
)
```

Arguments

genelist	a string vector, the gene list to be transferred.
now_name	a string, the current format of gene names, one of 'ensembl', 'symbol'.
to_name	a string, the format of gene names to transfer, one of 'ensembl', 'symbol'.
species	a string, the species, one of 'Human' and 'Mouse'.

Value

Return a string vector of transferred gene names. The gene names not matched in the database will not change.

Examples

```
## Not run:
geneNames <- c("ENSG00000171885", "ENSG00000115756")
transferGeneNames(geneNames, now_name = "ensembl", to_name="symbol", species="Human")

## End(Not run)
```

Index

* datasets

CosMx_subset, [7](#)
pbmc3k_subset, [22](#)
top5_signatures, [25](#)

AddAdj, [2](#)

AddParSettingFAST, [4](#)

coembed_plot, [4](#)

coembedding_umap, [6](#), [6](#)

CosMx_subset, [7](#)

CreatePRECASTObject, [4](#), [10](#), [17](#), [24](#)

diagnostic.cor.eigs, [8](#)

diagnostic.cor.eigs.default, [9](#)

FAST, [10](#), [12](#), [14](#)

FAST_run, [10](#), [12–14](#), [22](#)

FAST_single, [12](#)

FAST_structure, [12](#), [13](#)

find.signature.genes, [14](#), [15](#)

get.top.signature.dat, [15](#)

get_r2_mcfadden, [16](#)

getAdj_auto, [3](#)

IntegrateSRTData, [17](#)

iscmeb_run, [18](#), [24](#)

Mclust, [19](#)

model_set_FAST, [4](#), [12–14](#), [19](#)

NCFM, [20](#)

NCFM_fast, [21](#)

pbmc3k_subset, [22](#)

pdistance, [23](#)

RunHarmonyLouvain, [24](#)

RunISCMEB, [24](#)

SelectHKgenes, [25](#)

top5_signatures, [25](#)

transferGeneNames, [26](#)