Package: dyndimred (via r-universe)

September 17, 2024

Type Package

Title Dimensionality Reduction Methods in a Common Format

Version 1.0.4

Description Provides a common interface for applying dimensionality reduction methods, such as Principal Component Analysis ('PCA'), Independent Component Analysis ('ICA'), diffusion maps, Locally-Linear Embedding ('LLE'), t-distributed Stochastic Neighbor Embedding ('t-SNE'), and Uniform Manifold Approximation and Projection ('UMAP'). Has built-in support for sparse matrices.

License MIT + file LICENSE

Encoding UTF-8

LazyData true

Imports dynutils (>= 1.0.5), irlba, lmds, stats, tibble

Suggests testthat, diffusionMap, fastICA, igraph, lle, MASS, Matrix, RANN, Rtsne, smacof, uwot

RoxygenNote 7.1.1

Roxygen list(markdown = TRUE)

URL https://github.com/dynverse/dyndimred

BugReports https://github.com/dynverse/dyndimred/issues

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

RemoteUrl https://github.com/dynverse/dyndimred

RemoteRef HEAD

RemoteSha b85e9f42a78bcf44e4e5de31beefa56a9cecf604

Contents

---

dimred *Perform simple dimensionality reduction*

Description

Perform simple dimensionality reduction

Usage

```
dimred(x, method, ndim, ...)

dimred_dm_destiny(
  x,
  ndim = 2,
  distance_method = c("euclidean", "spearman", "cosine")
)

dimred_dm_diffusionmap(
  x,
  ndim = 2,
  distance_method = c("pearson", "spearman", "cosine", "euclidean", "chisquared",
    "hamming", "kullback", "manhattan", "maximum", "canberra", "minkowski")
)

dimred_ica(x, ndim = 3)

dimred_knn_fr(
  x,
  ndim = 2,
  lmds_components = 10,
  distance_method = c("pearson", "spearman", "cosine", "euclidean", "chisquared",
    "hamming", "kullback", "manhattan", "maximum", "canberra", "minkowski"),
  n_neighbors = 10
)

dimred_landmark_mds(
  x,
  ndim = 2,
  distance_method = c("pearson", "spearman", "cosine", "euclidean", "chisquared",
    "hamming", "kullback", "manhattan", "maximum", "canberra", "minkowski")
)

dimred_lle(x, ndim = 3)

dimred_mds(
  x,
```

```
  ndim = 2,
  distance_method = c("pearson", "spearman", "cosine", "euclidean", "chisquared",
    "hamming", "kullback", "manhattan", "maximum", "canberra", "minkowski")
)

dimred_mds_isomds(
  x,
  ndim = 2,
  distance_method = c("pearson", "spearman", "cosine", "euclidean", "chisquared",
    "hamming", "kullback", "manhattan", "maximum", "canberra", "minkowski")
)

dimred_mds_sammon(
  x,
  ndim = 2,
  distance_method = c("pearson", "spearman", "cosine", "euclidean", "chisquared",
    "hamming", "kullback", "manhattan", "maximum", "canberra", "minkowski")
)

dimred_mds_smacof(
  x,
  ndim = 2,
  distance_method = c("pearson", "spearman", "cosine", "euclidean", "chisquared",
    "hamming", "kullback", "manhattan", "maximum", "canberra", "minkowski")
)

dimred_pca(x, ndim = 2)

list_dimred_methods()
```

Arguments

x	Log transformed expression data, with rows as cells and columns as features
method	The name of the dimensionality reduction method to use
ndim	The number of dimensions
...	Any arguments to be passed to the dimensionality reduction method
distance_method	The name of the distance metric, see dynutils::calculate_distance
lmds_components	The number of lmds components to use. If NULL, LMDS will not be performed first. If this is a matrix, it is assumed it is a dimred for x.
n_neighbors	The size of local neighborhood (in terms of number of neighboring sample points).

Examples

```
library(Matrix)
```

```
x <- abs(Matrix::rsparsematrix(100, 100, .5))
dimred(x, "pca", ndim = 3)
dimred(x, "ica", ndim = 3)

if (interactive()) {
  dimred_dm_destiny(x)
  dimred_dm_diffusionmap(x)
  dimred_ica(x)
  dimred_landmark_mds(x)
  dimred_lle(x)
  dimred_mds(x)
  dimred_mds_isomds(x)
  dimred_mds_sammon(x)
  dimred_mds_smacof(x)
  dimred_pca(x)
  dimred_tsne(x)
  dimred_umap(x)
}
```

`dimred_tsne` *tSNE*

Description

tSNE

Usage

```
dimred_tsne(
  x,
  ndim = 2,
  perplexity = 30,
  theta = 0.5,
  initial_dims = 50,
  distance_method = c("pearson", "spearman", "cosine", "euclidean", "chisquared",
    "hamming", "kullback", "manhattan", "maximum", "canberra", "minkowski")
)
```

Arguments

`x`	Log transformed expression data, with rows as cells and columns as features
`ndim`	The number of dimensions
`perplexity`	numeric; Perplexity parameter (should not be bigger than 3 * perplexity < nrow(X) - 1, see details for interpretation)
`theta`	numeric; Speed/accuracy trade-off (increase for less accuracy), set to 0.0 for exact TSNE (default: 0.5)

initial_dims integer; the number of dimensions that should be retained in the initial PCA step (default: 50)

distance_method
The name of the distance metric, see dynutils::calculate_distance

See Also

Rtsne::Rtsne()

Examples

```
library(Matrix)
dataset <- abs(Matrix::rsparsematrix(100, 100, .5))
dimred_tsne(dataset, ndim = 3)
```

---

dimred_umap *UMAP*

Description

UMAP

Usage

```
dimred_umap(
  x,
  ndim = 2,
  distance_method = c("euclidean", "cosine", "manhattan"),
  pca_components = 50,
  n_neighbors = 15L,
  init = "spectral",
  n_threads = 1
)
```

Arguments

x Log transformed expression data, with rows as cells and columns as features

ndim The number of dimensions

distance_method
The name of the distance metric, see dynutils::calculate_distance

pca_components The number of pca components to use for UMAP. If NULL, PCA will not be performed first

n_neighbors The size of local neighborhood (in terms of number of neighboring sample points).

init Type of initialization for the coordinates. Options are:

- "spectral" Spectral embedding using the normalized Laplacian of the fuzzy 1-skeleton, with Gaussian noise added.
- "normlaplacian". Spectral embedding using the normalized Laplacian of the fuzzy 1-skeleton, without noise.
- "random". Coordinates assigned using a uniform random distribution between -10 and 10.
- "lvrandom". Coordinates assigned using a Gaussian distribution with standard deviation 1e-4, as used in LargeVis (Tang et al., 2016) and t-SNE.
- "laplacian". Spectral embedding using the Laplacian Eigenmap (Belkin and Niyogi, 2002).
- "pca". The first two principal components from PCA of X if X is a data frame, and from a 2-dimensional classical MDS if X is of class "dist".
- "spca". Like "pca", but each dimension is then scaled so the standard deviation is 1e-4, to give a distribution similar to that used in t-SNE. This is an alias for init = "pca", init_sdev = 1e-4.
- "agspectral" An "approximate global" modification of "spectral" which all edges in the graph to a value of 1, and then sets a random number of edges (negative_sample_rate edges per vertex) to 0.1, to approximate the effect of non-local affinities.
- A matrix of initial coordinates.

For spectral initializations, ("spectral", "normlaplacian", "laplacian"), if more than one connected component is identified, each connected component is initialized separately and the results are merged. If verbose = TRUE the number of connected components are logged to the console. The existence of multiple connected components implies that a global view of the data cannot be attained with this initialization. Either a PCA-based initialization or increasing the value of n_neighbors may be more appropriate.

n_threads Number of threads to use (except during stochastic gradient descent). Default is half the number of concurrent threads supported by the system. For nearest neighbor search, only applies if nn_method = "annoy". If n_threads > 1, then the Annoy index will be temporarily written to disk in the location determined by tempfile.

See Also

uwot::umap()

Examples

```
library(Matrix)
dataset <- abs(Matrix::rsparsematrix(100, 100, .5))
dimred_umap(dataset, ndim = 2, pca_components = NULL)
```

dyndimred *Common dimensionality reduction methods*

Description

Provides a common interface for applying common dimensionality reduction methods, Such as PCA, ICA, diffusion maps, LLE, t-SNE, and umap.

Index