

# Package: WormTensor (via r-universe)

August 23, 2024

**Type** Package

**Title** A Clustering Method for Time-Series Whole-Brain Activity Data of 'C. elegans'

**Version** 0.1.1

**Maintainer** Kentaro Yamamoto <yamaken37.the.answer@gmail.com>

**Description** A toolkit to detect clusters from distance matrices. The distance matrices are assumed to be calculated between the cells of multiple animals ('Caenorhabditis elegans') from input time-series matrices. Some functions for generating distance matrices, performing clustering, evaluating the clustering, and visualizing the results of clustering and evaluation are available. We're also providing the download function to retrieve the calculated distance matrices from 'figshare' <<https://figshare.com>>.

**License** MIT + file LICENSE

**Encoding** UTF-8

**URL** <https://github.com/rikenbit/WormTensor>

**BugReports** <https://github.com/rikenbit/WormTensor/issues>

**Suggests** knitr, rmarkdown, testthat (>= 3.0.0)

**Config/testthat/edition** 3

**Depends** R (>= 4.1.0)

**Imports** rTensor, usedist, dtwclust, clusterSim, clValid, aricode, cluster, ggplot2, Rtsne, uwot, factoextra, ggrepel, cowplot, methods

**RoxygenNote** 7.2.0

**VignetteBuilder** knitr

**NeedsCompilation** no

**Author** Kentaro Yamamoto [aut, cre], Koki Tsuyuzaki [aut], Itoshi Nikaido [aut]

**Repository** CRAN

**Date/Publication** 2024-07-23 09:10:02 UTC

## Contents

AllGenerics_component . . . . .	2
as_worm_tensor . . . . .	3
WormTensor-class . . . . .	4
worm_clustering,WormTensor-method . . . . .	4
worm_distance . . . . .	5
worm_download . . . . .	6
worm_evaluate,WormTensor-method . . . . .	7
worm_membership,WormTensor-method . . . . .	8
worm_visualize,WormTensor-method . . . . .	9

<b>Index</b>	<b>12</b>
--------------	-----------

---

AllGenerics\_component *Components for WormTensor object*

---

### Description

These are generic methods for WormTensor

### Usage

```
worm_membership(object, k)
```

```
worm_clustering(
  object,
  algorithm = c("MCMC", "OINDSCAL", "CSPA"),
  num.iter = 30,
  thr = 1e-10,
  verbose = FALSE
)
```

```
worm_evaluate(object, labels = NULL)
```

```
worm_visualize(
  object,
  out.dir = tempdir(),
  algorithm = c("tSNE", "UMAP"),
  seed = 1234,
  tsne.dims = 2,
  tsne.perplexity = 15,
  tsne.verbose = FALSE,
  tsne.max_iter = 1000,
  umap.n_neighbors = 15,
  umap.n_components = 2,
  silhouette.summary = FALSE
)
```

**Arguments**

object	WormTensor object
k	Assumed number of clusters
algorithm	Dimensional reduction methods
num.iter	The upper limit of iterations (Default value is 30)
thr	The lower limit of relative change in estimates (Default value is 1E-10)
verbose	Control message
labels	Labels for external evaluation
out.dir	Output directory (default: tempdir())
seed	Arguments passed to set.seed (default: 1234)
tsne.dims	Output dimensionality (default: 2)
tsne.perplexity	Perplexity paramete (default: 15)
tsne.verbose	logical; Whether progress updates should be printed (default: TRUE)
tsne.max_iter	The number of iterations (default: 1000)
umap.n_neighbors	The size of local neighborhood (default: 15)
umap.n_components	The dimension of the space to embed into (default: 2)
silhouette.summary	logical; If true a summary of cluster silhouettes are printed.

---

as_worm_tensor	<i>Generates WormTensor object A WormTensor object is generated from distance matrices.</i>
----------------	---

---

**Description**

Generates WormTensor object A WormTensor object is generated from distance matrices.

**Usage**

```
as_worm_tensor(Ds)
```

**Arguments**

Ds                    A list containing distance matrices

**Value**

An object containing distance matrices and metadata

**Examples**

```
worm_download("mSBD", qc = "PASS")$Ds |> as_worm_tensor() -> object
```

---

WormTensor-class      *S4 class used by as\_worm\_tensor.R, worm\_membership.R, worm\_clustering.R, worm\_evaluate.R, worm\_visualize.R*

---

### Description

S4 class used by as\_worm\_tensor.R, worm\_membership.R, worm\_clustering.R, worm\_evaluate.R, worm\_visualize.R

### Slots

dist\_matrices list  
n\_animals numeric  
union\_cellnames character  
n\_union\_cells numeric  
membership\_tensor Tensor  
k numeric  
clustering\_algorithm character  
clustering numeric  
weight numeric  
factor matrix  
consensus matrix  
eval list  
dimension\_reduction\_algorithm character

---

worm\_clustering, WormTensor-method  
*Generates clustering result A clustering result is generated from a membership tensor.*

---

### Description

Generates clustering result A clustering result is generated from a membership tensor.

### Usage

```
## S4 method for signature 'WormTensor'
worm_clustering(
  object,
  algorithm = c("MCFI", "OINDSCAL", "CSPA"),
  num.iter = 30,
  thr = 1e-10,
  verbose = FALSE
)
```

**Arguments**

object	WormTensor object with a membership tensor
algorithm	Clustering methods
num.iter	The upper limit of iterations (Default value is 30)
thr	The lower limit of relative change in estimates (Default value is 1E-10)
verbose	Control message

**Value**

WormTensor object with a clustering result added

**Examples**

```
# Pipe Operation
worm_download("Euclid", qc = "WARN")$Ds |>
  as_worm_tensor() |>
  worm_membership(k = 6) -> object
worm_clustering(object, verbose = TRUE) -> ob_mcmi
worm_clustering(object, algorithm = "OINDSCAL", verbose = TRUE) -> ob_oind
worm_clustering(object, algorithm = "CSPA", verbose = TRUE) -> ob_cspa
```

---

worm_distance	<i>Generates distance matrices Distance matrices are generated between the cells of multiple animals (Caenorhabditis elegans) from time-series matrices.</i>
---------------	--

---

**Description**

Generates distance matrices Distance matrices are generated between the cells of multiple animals (Caenorhabditis elegans) from time-series matrices.

**Usage**

```
worm_distance(data, distance = c("mSBD", "SBD", "Euclid"))
```

**Arguments**

data	Time-series matrices
distance	"mSBD" or "SBD" or "Euclid" can be specified. mSBD means modified Shape-based distance.

**Value**

A list containing distance matrices

**Examples**

```

# Toy data
n_cell_x <- 13
n_cell_y <- 24
n_cell_z <- 29
n_cells <- 30
n_time_frames <- 100

# 13 cells, 100 time frames
animal_x <- matrix(runif(n_cell_x * n_time_frames),
  nrow = n_cell_x, ncol = n_time_frames
)
rownames(animal_x) <- sample(seq(n_cells), n_cell_x)
colnames(animal_x) <- seq(n_time_frames)

# 24 cells, 100 time frames
animal_y <- matrix(runif(n_cell_y * n_time_frames),
  nrow = n_cell_y, ncol = n_time_frames
)
rownames(animal_y) <- sample(seq(n_cells), n_cell_y)
colnames(animal_y) <- seq(n_time_frames)

# 29 cells, 100 time frames
animal_z <- matrix(runif(n_cell_z * n_time_frames),
  nrow = n_cell_z, ncol = n_time_frames
)
rownames(animal_z) <- sample(seq(n_cells), n_cell_z)
colnames(animal_z) <- seq(n_time_frames)

# Positive Control of Difference between SBD and mSBD
animal_z[2, ] <- -animal_x[1, ]
X <- list(
  animal_x = animal_x,
  animal_y = animal_y,
  animal_z = animal_z
)
Ds_mSBD <- worm_distance(X, "mSBD")

```

---

worm\_download

*Downloads distance matrices 28 animals' data including 24 normal and 4 noisy are retrieved from figshare.*


---

**Description**

Downloads distance matrices 28 animals' data including 24 normal and 4 noisy are retrieved from figshare.

**Usage**

```
worm_download(distance = c("mSBD", "Euclid"), qc = c("PASS", "WARN", "FAIL"))
```

**Arguments**

distance "mSBD" or "Euclid" can be specified

qc "PASS" or "WARN" or "FAIL" can be specified. "PASS" downloads 24 data except 4 noisy data. "WARN" downloads 27 data except 1 noisy data. "FAIL" downloads all 28 data.

**Value**

A List of containing distance matrices. The list also includes metadata for each animals.

**Examples**

```
Ds_Euclid <- worm_download("Euclid", qc = "WARN")
Ds_mSBD <- worm_download("mSBD", qc = "PASS")
```

---

worm\_evaluate, WormTensor-method

*Evaluates clustering result An evaluation result is generated from a WormTensor object.*

---

**Description**

Evaluates clustering result An evaluation result is generated from a WormTensor object.

**Usage**

```
## S4 method for signature 'WormTensor'
worm_evaluate(object, labels = NULL)
```

**Arguments**

object WormTensor object with a result of worm\_clustering

labels Labels for external evaluation

**Value**

WormTensor object with an evaluation result added

**Examples**

```
# Pipe Operation
worm_download("mSBD", qc = "PASS")$Ds |>
  as_worm_tensor() |>
  worm_membership(k = 6) |>
  worm_clustering() -> object
# Internal evaluation
worm_evaluate(object) -> object_internal
```

```

# External evaluation by sample labels
labels <- list(
  label1 = sample(3, length(object@clustering), replace = TRUE),
  label2 = sample(4, length(object@clustering), replace = TRUE),
  label3 = sample(5, length(object@clustering), replace = TRUE)
)
worm_evaluate(object, labels) -> object_external

# External evaluation by worm_download labels
Ds_mSBD <- worm_download("mSBD", qc = "PASS")
labels <- list(
  label1 = replace(
    Ds_mSBD$labels$Class,
    which(is.na(Ds_mSBD$labels$Class)),
    "NA"
  ),
  label2 = sample(4, length(object@clustering), replace = TRUE),
  label3 = sample(5, length(object@clustering), replace = TRUE)
)
worm_evaluate(object, labels) -> object_external_Class

```

---

worm\_membership, WormTensor-method

*Generates membership tensor A membership tensor is generated from distance matrices.*

---

## Description

Generates membership tensor A membership tensor is generated from distance matrices.

## Usage

```
## S4 method for signature 'WormTensor'
worm_membership(object, k)
```

## Arguments

object	WormTensor object with distance matrices
k	Assumed number of clusters

## Value

WormTensor object with membership tensor added



**Examples**

```
# Pipe Operation
worm_download("mSBD", qc = "PASS")$Ds |>
  as_worm_tensor() -> object
# k=6
worm_membership(object, k = 6) -> object_k6
```

---

```
worm_visualize, WormTensor-method
```

*Plots evaluation result A visualization result is generated from a WormTensor object.*

---

**Description**

Plots evaluation result A visualization result is generated from a WormTensor object.

**Usage**

```
## S4 method for signature 'WormTensor'
worm_visualize(
  object,
  out.dir = tempdir(),
  algorithm = c("tSNE", "UMAP"),
  seed = 1234,
  tsne.dims = 2,
  tsne.perplexity = 15,
  tsne.verbose = FALSE,
  tsne.max_iter = 1000,
  umap.n_neighbors = 15,
  umap.n_components = 2,
  silhouette.summary = FALSE
)
```

**Arguments**

object	WormTensor object with a result of worm_evaluate
out.dir	Output directory (default: tempdir())
algorithm	Dimensional reduction methods
seed	Arguments passed to set.seed (default: 1234)
tsne.dims	Output dimensionality (default: 2)
tsne.perplexity	Perplexity parameter (default: 15)
tsne.verbose	logical; Whether progress updates should be printed (default: TRUE)
tsne.max_iter	Number of iterations (default: 1000)

```

umap.n_neighbors
    The size of the local neighborhood (default: 15)
umap.n_components
    The dimension of the space to embed into (default: 2)
silhouette.summary
    logical; If true a summary of cluster silhouettes are printed.

```

### Value

Silhouette plots. ARI with a merge result and each animal(with MCMI). Dimensional reduction plots colored by cluster, no. of identified cells, consistency(with labels), Class\_label(with labels).

### References

The `.dist_nn` function is quoted from `dist_nn` (not exported function) in package `uwot`(<https://github.com/jlmlville/uwot/tree/f467185c8cbcd158feb60dde608c9da153ed10d7>).

### Examples

```

# Temporary directory to save figures
out.dir <- tempdir()

# Labels
worm_download("mSBD", qc = "PASS")$Ds |>
  as_worm_tensor() |>
  worm_membership(k = 6) |>
  worm_clustering() -> object
Ds_mSBD <- worm_download("mSBD", qc = "PASS")
labels <- list(
  label1 = replace(
    Ds_mSBD$labels$Class,
    which(is.na(Ds_mSBD$labels$Class)),
    "NA"
  ),
  label2 = sample(4, length(object@clustering), replace = TRUE),
  label3 = sample(5, length(object@clustering), replace = TRUE)
)

# Pipe Operation (without Labels)
worm_download("mSBD", qc = "PASS")$Ds |>
  as_worm_tensor() |>
  worm_membership(k = 6) |>
  worm_clustering() |>
  worm_evaluate() |>
  worm_visualize(out.dir) -> object_no_labels

# Pipe Operation (with Labels)
worm_download("mSBD", qc = "PASS")$Ds |>
  as_worm_tensor() |>
  worm_membership(k = 6) |>
  worm_clustering() |>
  worm_evaluate(labels) |>

```

```
worm_visualize(out.dir) -> object_labels
```

# Index

AllGenerics\_component, [2](#)  
as\_worm\_tensor, [3](#)

worm\_clustering  
    ([AllGenerics\\_component](#)), [2](#)  
worm\_clustering, [WormTensor-method](#), [4](#)  
worm\_distance, [5](#)  
worm\_download, [6](#)  
worm\_evaluate ([AllGenerics\\_component](#)), [2](#)  
worm\_evaluate, [WormTensor-method](#), [7](#)  
worm\_membership  
    ([AllGenerics\\_component](#)), [2](#)  
worm\_membership, [WormTensor-method](#), [8](#)  
worm\_visualize ([AllGenerics\\_component](#)),  
    [2](#)  
worm\_visualize, [WormTensor-method](#), [9](#)  
[WormTensor-class](#), [4](#)