Package 'TMixClust'

April 12, 2018

Type Package

Title Time Series Clustering of Gene Expression with Gaussian Mixed-Effects Models and Smoothing Splines

Version 1.0.1

Year 2017

Date 2017-06-04

Author Monica Golumbeanu <golumbeanu.monica@gmail.com>

Maintainer Monica Golumbeanu <golumbeanu.monica@gmail.com>

Description Implementation of a clustering method for time series gene expression data based on mixed-effects models with Gaussian variables and non-parametric cubic splines estimation. The method can robustly account for the high levels of noise present in typical gene expression time series datasets.

License GPL (>=2)

Depends R (>= 3.4)

biocViews Software, StatisticalMethod, Clustering, TimeCourse, GeneExpression

Imports gss, mvtnorm, stats, zoo, cluster, utils, BiocParallel, flexclust, grDevices, graphics, Biobase, SPEM

Suggests rmarkdown, knitr, BiocStyle, testthat

VignetteBuilder knitr

Encoding UTF-8

LazyData true

RoxygenNote 6.0.1

NeedsCompilation no

R topics documented:

`TMixClust-package` *The main usages of TMixClust*

Description

Description of package TMixClust

Overview

TMixClust is a soft-clustering method which employs mixed-effects models with nonparametric smoothing spline fitting and is able to robustly stratify genes by their complex time series patterns. The package has, besides the main clustering method, a set of functionalities assisting the user to visualise and assess the clustering results, and to choose the most optimal clustering solution.

Author(s)

Monica Golumbeanu, <monica.golumbeanu@bsse.ethz.ch>

References

Golumbeanu M, Desfarges S, Hernandez C, Quadroni M, Rato S, Mohammadi P, Telenti A, Beerenwinkel N, Ciuffi A. (2017) Dynamics of Proteo-Transcriptomic Response to HIV-1 Infection.

`analyse_stability` *Stability analysis, clustering evaluation and optimal solution selection*

Description

`analyse_stability` Performs multiple clustering runs with TMixClust, analyses the agreement between runs with the Rand index and returns the clustering solution with the largest likelihood. A plot of agreement probability between all the runs and the run with the maximum likelihood is produced.

Usage

```
analyse_stability(time_series_df, time_points = seq_len(ncol(time_series_df)),
  nb_clusters = 2, em_iter_max = 1000, mc_em_iter_max = 10,
  em_ll_convergence = 0.001, nb_clustering_runs = 3, nb_cores = 1)
```

Arguments

time_series_df	data frame containing the time series. Each row is a time series comprised of the time series name which is also the row name, and the time series values at each time point.
time_points	vector containing numeric values for the time points. Default: seq_len(ncol(time_series_df)).
nb_clusters	desired number of clusters
em_iter_max	maximum number of iterations for the expectation-maximization (EM) algorithm. Default: 1000.
mc_em_iter_max	maximum number of iterations for Monte-Carlo resampling. Default is 10.
em_ll_convergence	convergence threshold for likelihood improvement. Default is 0.001.
nb_clustering_runs	number of times the clustering procedure is repeated on the input data. Default is 3.
nb_cores	number of cores to be used to run the separate clustering operations in parallel. Default is 1.

Value

TMixClust object with the highest likelihood. Renders a plot showing the overall distribution of the Rand index, which allows the user to assess clustering stability.

Author(s)

Monica Golumbeanu, <monica.golumbeanu@bsse.ethz.ch>

References

Golumbeanu M, Desfarges S, Hernandez C, Quadroni M, Rato S, Mohammadi P, Telenti A, Beerenwinkel N, Ciuffi A. (2017) Dynamics of Proteo-Transcriptomic Response to HIV-1 Infection.

Examples

```
# Load the toy time series data provided with the TMixClust package
data(toy_data_df)

# Identify the most optimal clustering solution with 3 clusters
best_clust_obj = analyse_stability(toy_data_df, nb_clusters = 3,
                                   nb_clustering_runs = 4, nb_cores = 1)

# Plot the time series from each cluster
for (i in seq_len(3)) {
    # Extract the time series in the current cluster and plot them
    c_df=toy_data_df[which(best_clust_obj$em_cluster_assignment==i),]
    plot_time_series_df(c_df, plot_title = paste("cluster",i))
}
```

`best_clust_toy_obj` *TMixClust object containing the optimal clustering solution for the toy data with 3 clusters.*

Description

This object contains the result of clustering and stability analysis corresponding to the clustering solution with the highest likelihood among 10 different runs of clustering on the toy data with K=3 clusters.

Usage

```
best_clust_toy_obj
```

Format

A `TMixClust` object.

Value

optimal clustering solution for the toy data

Author(s)

Monica Golumbeanu, <monica.golumbeanu@bsse.ethz.ch>

References

Golumbeanu M, Desfarges S, Hernandez C, Quadroni M, Rato S, Mohammadi P, Telenti A, Beerenwinkel N, Ciuffi A. (2017) Dynamics of Proteo-Transcriptomic Response to HIV-1 Infection.

Examples

```
# Load the optimal clustering solution for the toy data
# provided with the TMixClust package
data("best_clust_toy_obj")

# Print the first lines of the toy clustering object
head(best_clust_toy_obj)
```

`best_clust_yeast_obj` *TMixClust object containing the optimal clustering solution for the yeast data.*

Description

This object contains the result of clustering and stability analysis corresponding to the clustering solution with the highest likelihood among 10 different runs of clustering on the yeast data with K=4 clusters.

Usage

```
best_clust_yeast_obj
```

Format

A `TMixClust` object.

Value

optimal clustering solution for the yeast data

Author(s)

Monica Golumbeanu, <monica.golumbeanu@bsse.ethz.ch>

References

Golumbeanu M, Desfarges S, Hernandez C, Quadroni M, Rato S, Mohammadi P, Telenti A, Beerenwinkel N, Ciuffi A. (2017) Dynamics of Proteo-Transcriptomic Response to HIV-1 Infection.

Examples

```
# Load the optimal clustering solution for the yeast data
# provided with the TMixClust package
data("best_clust_yeast_obj")

# Print the first lines of the yeast clustering object
head(best_clust_yeast_obj)
```

---

generate_TMixClust_report

Generates a series of files containing a summary of the TMixClust analysis results

---

Description

`generate_TMixClust_report`

Usage

```
generate_TMixClust_report(TMixClust_object, report_folder = paste(getwd(),
  "/TMixClust_report/", sep = ""), data_color = "#fd8d3c", x_label = "time",
  y_label = "value")
```

Arguments

TMixClust_object	list object created by the TMixClust function (see function TMixClust)
report_folder	full path of the folder where the report files will be saved. Default is TMixClust_report/ folder in current working directory.
data_color	color of the time series to be used when generating the cluster plots. Default is orange.
x_label	label of the x axis for the cluster plots. Default is "time"
y_label	label of the y axis for the cluster plots. Default is "value"

Value

Produces a series of files containing information about the clustering results and saves them in the provided folder location. The folder contains the following:

- log-lihelihood.txt - file with the log likelihood values at each iteration on separate lines
- log-likelihood.pdf - plot of log-likelihood at each iteration
- posterior.txt - file with the posterior probabilities of all the time-series for each cluster
- estimated_curves/ - folder containing a number of files equal to the number of clusters; each file has 4 lines consisting of curve values and their confidence intervals (first 3 lines) for a discrete time grid (last line).
- clusters/ - folder containing a plot with the time series in each cluster, a silhouette plot of the clustering configuration, as well as, for each cluster, a file containing the names of the time series in the respective cluster and a file containing the names and time series values for the time series in each cluster.

Author(s)

Monica Golumbeanu, <monica.golumbeanu@bsse.ethz.ch>

References

Golumbeanu M, Desfarges S, Hernandez C, Quadroni M, Rato S, Mohammadi P, Telenti A, Beerenwinkel N, Ciuffi A. (2017) Dynamics of Proteo-Transcriptomic Response to HIV-1 Infection.

Examples

```
## Not run:
# Load the toy time series data provided with the TMixClust package
data(toy_data_df)

# Cluster the toy data with default parameters
TMixClust_obj = TMixClust(toy_data_df)

# Generate a TMixClust report in the current working directory
generate_TMixClust_report(TMixClust_obj)

## End(Not run)
```

get_time_series_df *Extracts a time series data frame from a text file*

Description

get_time_series_df creates a data frame containing time series data from a file.

Usage

```
get_time_series_df(data_file)
```

Arguments

data_file	path to a tab-delimited text file containing the time series data formatted such that each row contains a time-series represented by its name (e.g. gene name, protein name, etc.) and the values at each time point.

Value

A data frame containing the time series

Author(s)

Monica Golumbeanu, <monica.golumbeanu@bsse.ethz.ch>

References

Golumbeanu M, Desfarges S, Hernandez C, Quadroni M, Rato S, Mohammadi P, Telenti A, Beerenwinkel N, Ciuffi A. (2017) Dynamics of Proteo-Transcriptomic Response to HIV-1 Infection.

Examples

```
# Load a simulated toy time-series data provided with the package
toy_data_file = system.file("extdata", "toy_time_series.txt",
package = "TMixClust")
toy_data= get_time_series_df(toy_data_file)

# Print the first lines of the resulting data frame
print(head(toy_data))
```

get_time_series_df_bio *Extracts a time series data frame from a Bioconductor Biobase ExpressionSet object.*

Description

get_time_series_df_bio creates a data frame with time series data from a Bioconductor Biobase ExpressionSet object.

Usage

```
get_time_series_df_bio(bio_obj)
```

Arguments

bio_obj	Bioconductor Biobase ExpressionSet object. The assayData has to contain a matrix where each row is a gene time series and each column contains the time series values at each time point. The number of columns is equal to the number of time points, while the number of rows is equal to the number of genes.

Value

A data frame containing the time series

Author(s)

Monica Golumbeanu, <monica.golumbeanu@bsse.ethz.ch>

References

Golumbeanu M, Desfarges S, Hernandez C, Quadroni M, Rato S, Mohammadi P, Telenti A, Beerenwinkel N, Ciuffi A. (2017) Dynamics of Proteo-Transcriptomic Response to HIV-1 Infection.

Examples

```
# Load the SOS pathway data from Bioconductor package SPEM
library(SPEM)
data(sos)
sos_data = get_time_series_df_bio(sos)

# Print the first lines of the retrieved time series data frame
print(head(sos_data))
```

---

plot_silhouette *Generates a silhouette plot for a given clustering configuration.*

Description

plot_silhouette

Usage

```
plot_silhouette(TMixClust_object, sim_metric = "euclidean",
  sil_color = "#bdbdbd")
```

Arguments

TMixClust_object	list object created by the TMixClust function (see function TMixClust)
sim_metric	character string taking one of the possible values: "euclidean", "gower" or "manhattan". Default is "euclidean".
sil_color	color of the bars representing the silhouette widths on the plot

Value

List object with the following components:

- `similarity_m` similarity matrix
- `silh` silhouette object

Renders a plot comprised of a set of barplots with the distributions of silhouette coefficients for the data points in each cluster. Each barplot has indicated on its right hand side the total number of points in the corresponding cluster. The plot also indicates with a dotted line, the overall average silhouette width, whose value is specified at the bottom of the plot.

Author(s)

Monica Golumbeanu, <monica.golumbeanu@bsse.ethz.ch>

References

Golumbeanu M, Desfarges S, Hernandez C, Quadroni M, Rato S, Mohammadi P, Telenti A, Beerenwinkel N, Ciuffi A. (2017) Dynamics of Proteo-Transcriptomic Response to HIV-1 Infection.

Examples

```
# Load the TMixClust object associated to the toy time series data
# provided with the TMixClust package
data(best_clust_toy_obj)

# Plot the silhouette for the clustering stored in the toy TMixClust object
plot_silhouette(best_clust_toy_obj)
```

`plot_time_series_df` *Plots all the time series stored in a data frame object*

Description

`plot_time_series_df` allows the user to visualise the time series from a given data set.

Usage

```
plot_time_series_df(ts_df, time_points = seq_len(ncol(ts_df)),
  data_color = "#fd8d3c", x_label = "time", y_label = "value",
  plot_title = "Time series plot")
```

Arguments

`ts_df`	data frame containing on each row a time-series
`time_points`	vector containing the values of the time points. Default: `seq_len(ncol(time_series_df))`.
`data_color`	color of the time series to be used for the plot. Default is orange.
`x_label`	label of the x axis of the plot. Default is "time"
`y_label`	label of the y axis of the plot. Default is "value"
`plot_title`	title of the plot. Default is "Time series plot".

Value

Plots a figure with all the the time series in the data set

Author(s)

Monica Golumbeanu, <monica.golumbeanu@bsse.ethz.ch>

References

Golumbeanu M, Desfarges S, Hernandez C, Quadroni M, Rato S, Mohammadi P, Telenti A, Beerenwinkel N, Ciuffi A. (2017) Dynamics of Proteo-Transcriptomic Response to HIV-1 Infection.

Examples

```
# Load the toy time series data provided with the TMixClust package
data(toy_data_df)

# Plot the time series
plot_time_series_df(toy_data_df)
```

`TMixClust` *Clusters the time series data in a given number of groups*

Description

`TMixClust` is the central function of the package. It clusters the given time series data into a specified number of clusters.

Usage

```
TMixClust(time_series_df, time_points = seq_len(ncol(time_series_df)),
  nb_clusters = 2, em_iter_max = 1000, mc_em_iter_max = 10,
  em_ll_convergence = 0.001)
```

Arguments

`time_series_df`	data frame containing the time series. Each row is a time series comprised of the time series name which is also the row name, and the time series values at each time point.
`time_points`	vector containing numeric values for the time points. Default: `seq_len(ncol(time_series_df))`.
`nb_clusters`	desired number of clusters
`em_iter_max`	maximum number of iterations for the expectation-maximization (EM) algorithm. Default: 1000.
`mc_em_iter_max`	maximum number of iterations for Monte-Carlo resampling. Default is 100.
`em_ll_convergence`	convergence threshold for likelihood improvement. Default is 0.001.

Value

list object with the following attributes:

- `em_gss_obj_list` object of class `gss` containing estimated parameters of the mixed-effects model (see package vignette for more details).
- `em_pi_k` vector containing the mixing coefficients corresponding to each cluster
- `em_mat_post` matrix containing the posterior values for each time series and cluster
- `em_cluster_assignment` vector with the clustering attribution for each time series
- `el_ll` vector containing the log likelihood values at each iteration in the EM algorithm
- `ts_data` the same as the input time series data-frame
- `ts_time_points` the same as the input time-points vector

Author(s)

Monica Golumbeanu, <monica.golumbeanu@bsse.ethz.ch>

References

Golumbeanu M, Desfarges S, Hernandez C, Quadroni M, Rato S, Mohammadi P, Telenti A, Beerenwinkel N, Ciuffi A. (2017) Dynamics of Proteo-Transcriptomic Response to HIV-1 Infection.

Examples

```
# Load the toy time series data provided with the TMixClust package
data(toy_data_df)

# Cluster the toy data with default parameters
TMixClust_obj = TMixClust(toy_data_df)
```

---

toy_data_df *Simulated time-series gene expression data*

Description

This data set contains a toy example of time-series gene expression data.

Usage

```
toy_data_df
```

Format

A `data frame` with 91 rows and 6 columns. The columns correspond to different time points, while the rownames of the data frame correspond to gene names.

Value

toy data

Author(s)

Monica Golumbeanu, <monica.golumbeanu@bsse.ethz.ch>

References

Golumbeanu M, Desfarges S, Hernandez C, Quadroni M, Rato S, Mohammadi P, Telenti A, Beerenwinkel N, Ciuffi A. (2017) Dynamics of Proteo-Transcriptomic Response to HIV-1 Infection.

Examples

```
# Load the toy time series data provided with the TMixClust package
data("toy_data_df")

# Print the first lines of the toy data frame
head(toy_data_df)
```

Index