Package: expSBM (via r-universe)

September 18, 2024

Type Package

Title An Exponential Stochastic Block Model for Interaction Lengths

Version 1.3.5

Date 2019-10-15

Description Given a continuous-time dynamic network, this package allows one to fit a stochastic blockmodel where nodes belonging to the same group create interactions and non-interactions of similar lengths. This package implements the methodology described by R. Rastelli and M. Fop (2019) <arXiv:1901.09828>.

License GPL-3

Imports Rcpp (>= 0.12.10), mclust, gtools, blockmodels

LinkingTo Rcpp, RcppArmadillo

LazyData TRUE

NeedsCompilation yes

Author Riccardo Rastelli [aut, cre] (<`https://orcid.org/0000-0003-0982-2935`>), Michael Fop [aut] (<`https://orcid.org/0000-0003-3936-2757`>)

Maintainer Riccardo Rastelli <`riccardoras@gmail.com`>

Date/Publication 2019-10-15 18:10:02 UTC

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

RemoteUrl https://github.com/cran/expSBM

RemoteRef HEAD

RemoteSha 1a78ef7ff613694bc22eaefcf2a3536c5bf50f71

Contents

`expSBM-package` *An Exponential Stochastic Block Model for Interaction Lengths*

Description

Given a continuous-time dynamic network, this package allows one to fit a stochastic blockmodel where nodes belonging to the same group create interactions and non-interactions of similar lengths. This package implements the methodology described by R. Rastelli and M. Fop (2019) <arXiv:1901.09828>.

Author(s)

Riccardo Rastelli <riccardoras@gmail.com>

Michael Fop <michael.fop@ucd.ie>

References

R. Rastelli and M. Fop (2019) "A dynamic stochastic blockmodel for interaction lengths", `https://arxiv.org/abs/1901.09828`

`expSBM_ELBO` *expSBM_ELBO*

Description

Evaluates the evidence lower bound for a given dynamic network.

Usage

```
expSBM_ELBO(N, edgelist, Z, lambda, mu, nu, directed = F, trunc = T, verbose = F)
```

Arguments

`N`	Number of nodes.
`edgelist`	A matrix with 4 columns: on the first column the sender node, on the second the receiver, on the third either a one or zero to indicate whether it is an interaction or a non-interaction respectively, on the fourth the corresponding exponential length.
`Z`	A `NxK` matrix indicating a soft clustering of the nodes into the `K` latent groups. The generic entry in position `[i,k]` represents the posterior probability that node `i` belongs to group `k`.
`lambda`	Mixing proportions of the latent groups.
`mu`	A matrix of size `KxK` indicating the exponential rates for the interaction lengths, for each pair of groups. Must be a symmetric matrix if `directed` is false.

`nu`	A matrix of size `KxK` indicating the exponential rates for the non-interaction lengths, for each pair of groups. Must be a symmetric matrix if `directed` is false.
`directed`	`TRUE` or `FALSE` indicating whether interactions have an orientation or not.
`trunc`	`TRUE` or `FALSE` indicating whether the first and last interactions or non-interactions for every edge are assumed to be truncated or not.
`verbose`	`TRUE` or `FALSE` indicating whether a lengthy output should be printed out.

Value

`computing_time`	Number of seconds required for the evaluation.
`elbo_value`	Value of the evidence lower bound for the given variational parameters.

`expSBM_EM` *expSBM_EM*

Description

Runs the variational expectation maximization algorithm for a given number of latent groups.

Usage

```
expSBM_EM(N, edgelist, Z, lambda, mu, nu, directed = F, trunc = T,
          tol = 0.001, n_iter_max = 100, verbose = F)
```

Arguments

`N`	Number of nodes.
`edgelist`	A matrix with 4 columns: on the first column the sender node, on the second the receiver, on the third either a one or zero to indicate whether it is an interaction or a non-interaction respectively, on the fourth the corresponding exponential length.
`Z`	A `NxK` matrix indicating a soft clustering of the nodes into the `K` latent groups. The generic entry in position `[i,k]` represents the posterior probability that node `i` belongs to group `k`.
`lambda`	Mixing proportions of the latent groups.
`mu`	A matrix of size `KxK` indicating the exponential rates for the interaction lengths, for each pair of groups. Must be a symmetric matrix if `directed` is false.
`nu`	A matrix of size `KxK` indicating the exponential rates for the non-interaction lengths, for each pair of groups. Must be a symmetric matrix if `directed` is false.
`directed`	`TRUE` or `FALSE` indicating whether interactions have an orientation or not.
`trunc`	`TRUE` or `FALSE` indicating whether the first and last interactions or non-interactions for every edge are assumed to be truncated or not.

`tol`	Stop the maximization if the relative increase in the objective function is not larger than this value.
`n_iter_max`	Stop the maximization if the number of iterations is larger than this value. This parameter can be set to zero or one for debug purposes.
`verbose`	`TRUE` or `FALSE` indicating whether a lengthy output should be printed out.

Value

`computing_time`	Number of seconds required for the evaluation.
`elbo_values`	Stored values of the objective function at each iteration.
`Z_star`	Optimal soft clustering of the nodes into the groups.
`lambda_star`	Optimal mixing proportions.
`mu_star`	Optimal group-specific parameters for the exponential rates of the interaction lengths.
`nu_star`	Optimal group-specific parameters for the exponential rates of the non-interaction lengths.

Examples

```
set.seed(1)
data(high_school)
K <- 4
lambda_init <- rep(1/K,K)
Z_init <- expSBM_init(high_school$edgelist, K, "random")
mu_init <- nu_init <- matrix(1,K,K)
expSBM_EM(N = 327, high_school$edgelist, Z_init, lambda_init, mu_init, nu_init)
```

`expSBM_init` *expSBM_init*

Description

Initialization step for the variational expectation maximization algorithm.

Usage

```
expSBM_init(edgelist, K,
   method = c("random", "SBM_binary", "SBM_gaussian", "spectral"),
   sbm_density = 0.5, blur_value = 1)
```

Arguments

`edgelist`	A matrix with 4 columns: on the first column the sender node, on the second the receiver, on the third either a one or zero to indicate whether it is an interaction or a non-interaction respectively, on the fourth the corresponding exponential length.
`K`	Number of latent groups.
`method`	Method used to initialise the allocations. Can be one of `random`, `SBM_binary`, `SBM_gaussian` or `spectral`. See details.
`sbm_density`	If `method == "SBM_binary"`, this is the target density for the thresholded binary stochastic blockmodel.
`blur_value`	A value between 0 and 1. If 1, the initialization method returns a hard partition where each node belongs to one group and one only. Reducing this value introduces noise, i.e. it gradually transforms the hard clustering into a soft clustering where each node is equally likely to belong to any of the K given clusters.

Details

All initialisation methods return a `NxK` matrix indicating the partitioning of the nodes.

The method `random` intialises the allocation variables uniformly at random.

The method `SBM_binary` first calculates the aggregated interaction and non-interaction times for each pair of nodes. Then, it calculates the portion of time when the nodes where interacting over the whole time period. Then it obtains an adjacency matrix by thresholding these values, i.e. values above a given threshold are replaced by ones and values below the threshold are replaced by zeros. The threshold is chosen by setting the parameter `sbm_density` which defines the desired density of the graph. Once the adjacency matrix is obtained, a binary stochastic blockmodel is fit on the data hence obtaining the partition.

The method `SBM_gaussian` aggregates the interaction values and non-interaction values for each pair of nodes. Then it log-transforms both of these quantities. Then it fits a stochastic blockmodel with multivariate Gaussian edges to obtain the partition.

The method `spectral` first calculates the aggregated interaction and non-interaction times for each pair of nodes. Then, it calculates the portion of time when the nodes where interacting over the whole time period. Then it performs model-based clustering on the Laplacian associated to this weighted matrix.

Value

A `NxK` matrix indicating the partitioning of the nodes.

Examples

```
set.seed(12345)
data(high_school)
K <- 4
lambda_init <- rep(1/K,K)
expSBM_init(high_school$edgelist, K, "random")
```

`expSBM_select` *expSBM_select*

Description

Runs the variational expectation maximization algorithm for different numbers of latent groups, and selects the best overall model using the integrated completed likelihood criterion. See reference for a detailed explanation of the procedure.

Usage

```
expSBM_select(K_max, N, edgelist, method = "SBM_gaussian", directed = F,
              trunc = T, tol = 0.001, n_iter_max = 100,
              init_blur_value = 1, verbose = F)
```

Arguments

`K_max`	Estimate and compare the models with number of latent groups equal to 1,2,...,`K_max`.
`N`	Number of nodes.
`edgelist`	A matrix with 4 columns: on the first column the sender node, on the second the receiver, on the third either a one or zero to indicate whether it is an interaction or a non-interaction respectively, on the fourth the corresponding exponential length.
`method`	Indicates the method used for the initialisation. Can be one of `random`, `SBM_binary`, `SBM_gaussian` or `spectral`. See `expSBM_init` for more details.
`directed`	`TRUE` or `FALSE` indicating whether interactions have an orientation or not.
`trunc`	`TRUE` or `FALSE` indicating whether the first and last interactions or non-interactions for every edge are assumed to be truncated or not.
`tol`	Stop the maximization if the relative increase in the objective function is not larger than this value.
`n_iter_max`	Stop the maximization if the number of iterations is larger than this value. This parameter can be set to zero or one for debug purposes.
`init_blur_value`	A value from zero to one, indicating if the initialized partition should be perturbed with noise. The value one means no noise, whereas the value zero has maximum noise, i.e. each node is equally likely belonging to any of the K groups.
`verbose`	`TRUE` or `FALSE` indicating whether a lengthy output should be printed out.

Value

`fitted_models`	A list with the fitted values for every model considered.
`icl_values`	Integrated completed likelihood values for each model considered.

`K_star`	Optimal number of latent groups, according to the integrated completed likelihood criterion.
`best_model`	Output of the variational expectation maximization algorithm for the best overall model.

References

R. Rastelli and M. Fop (2019) "A dynamic stochastic blockmodel for interaction lengths", https://arxiv.org/abs/1901.09828

Examples

```
set.seed(1)
data(high_school)

res <- expSBM_select(K_max = 8, N = 327, edgelist = high_school$edgelist,
     method = "random", tol = 0.01)
```

`high_school` *Interactions between high school students*

Description

The data concern face to face interactions among 327 high school students in Marseilles, France, and were collected by means of wearable sensors over a period of 5 days in December 2013. Students wore a sensor badge on their chest and the instrument recorded when they were facing each other with a time resolution of 20 seconds. Thus, any pair of students was considered interacting face-to-face when the sensors of the two were exchanging data packets at any given time during the 20 seconds interval. Additional information on the students is available from the same dataset. Students may have 4 different main specializations: biology (BIO), mathematics and physics (MP), physics and chemistry (PC), and engineering studies (PSI).

Usage

```
data(high_school)
```

Format

The list contains:

adj An adjacency list indicating whether any pair of students had at least one interaction during the 5 days of the study.

edgelist An edgelist in a format that can be handled by this package.

program Clustering variable indicating the program each student is registered to.

program_levels Names of the different programs.

program_aggr Aggregated version of the previous clustering variable, where programs are aggregated into 4 areas.

program_levels_aggr Names of the different areas, these correspond to biology (BIO), mathematics and physics (MP), physics and chemistry (PC), and engineering studies (PSI)

sex Clustering given by the sex of the students.

sex_levels Labels for each of the sex classes.

References

R. Mastrandrea, J. Fournet, and A. Barrat (2015). "Contact patterns in a high school: A comparison between data collected using wearable sensors, contact diaries and friendship surveys". *PLOS ONE 10.9, pp. 1-26.*

Index