Package: beam (via r-universe)

October 23, 2026

Type Package

Title Fast Bayesian Inference in Large Gaussian Graphical Models

Version 2.0.4

Date 2025-03-14

Author Gwenael G.R. Leday [cre, aut], Ilaria Speranza [aut], Harry Gray [ctb]

Maintainer Gwenael G.R. Leday <gwenael.leday@proton.me>

Depends R (>= 4.1.0)

Imports stats, methods, grDevices, graphics, Matrix, fdrtool, igraph, knitr, Rcpp, assertthat

Description Fast Bayesian inference of marginal and conditional independence structures from high-dimensional data. Leday and Richardson (2019), Biometrics, <doi:10.1111/biom.13064>.

LazyLoad yes

License GPL (>= 2.0)

LazyData TRUE

NeedsCompilation yes

RoxygenNote 7.3.2

LinkingTo Rcpp, RcppArmadillo, BH

Suggests covr, testthat

URL https://github.com/gleday/beam

Encoding UTF-8

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

Date/Publication 2025-04-11 10:10:07 UTC

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

RemoteRef HEAD

RemoteSha 99b744125702f549182ecb4f531fd4fbbecee798

Contents

beam-package 2
beam 3
beam-class 5
beam.select 7
beam.select-class 8
lightbeam 10
TCPAprad 11

Index **12**

`beam-package` *Fast Bayesian Inference in Large Gaussian Graphical Models*

Description

The package enables inference of marginal and conditional dependencies from high-dimensional using the method of Leday and Richardson (2019). Inference is carried out by multiple testing of hypotheses about pairwise (marginal or conditional) independence using closed-form Bayes factors. Exact tail probabilities are obtained from the null distributions of the Bayes factors to help address the multiplicity problem and control desired error rates for incorrect edge inclusion. The method is computationally very efficient and allows to address problems with hundreds or thousands of variables.

Details

1. The main function of the package is `beam` which carries out shrinkage estimation of the (inverse) covariance and compute the (scaled) Bayes factors as well as the tail probabilities (p-values). The function returns an (S4) object of class `beam-class` that is associated with the following methods:

- `summary,beam-method`:
provides a summary of inferred (marginal and/or conditional) associations.

- `marg,beam-method`:
returns a data.frame with marginal correlations, Bayes factors and/or tail probabilities.

- `cond,beam-method`:
returns a data.frame with partial correlations, Bayes factors and/or tail probabilities.

- `mcor,beam-method`:
return marginal correlation matrix (scaled posterior expectation of the covariance matrix).

- `pcor,beam-method`:
return partial correlation matrix (scaled posterior expectation of the inverse covariance matrix).

- `plotML,beam-method`:
plot log-marginal likelihood of the Gaussian conjugate model as a function of shrinkage parameter.

- `plotCor,beam-method`:
plot heatmap of marginal (upper triangle) and/or partial (lower triangle) correlation estimates.

2. The function beam.select takes as input an object of class beam-class and carries out edge selection by multiple testing of hypotheses about pairwise (marginal or conditional) independence. The function helps address the multiplicity problem and control different types of error rates (e.g. false discovery rate, family-wise error rate, ...). beam.select returns an (S4) object of class beam.select-class that is associated with the following methods:

- summary,beam.select-method:
provides a summary of inferred (marginal or conditional) associations.

- marg,beam.select-method:
returns a data.frame with marginal correlations, Bayes factors and/or tail probabilities for selected edges.

- cond,beam.select-method:
returns a data.frame with partial correlations, Bayes factors and/or tail probabilities for selected edges.

- bgraph,beam.select-method:
return an igraph object containing the marginal (in)dependence graph.

- ugraph,beam.select-method:
return an igraph object containing the conditional (in)dependence graph.

Author(s)

Authors: Gwenael G.R. Leday and Ilaria Speranza

Maintainer: Gwenael G.R. Leday <gwenael.leday@proton.me>

References

Leday, G.G.R. and Richardson, S. (2019). Fast Bayesian inference in large Gaussian graphical models. *Biometrics*. 75(4), 1288–1298.

---

beam *Bayesian inference in large Gaussian graphical models*

Description

This function carries out covariance and inverse covariance estimation within the Gaussian conjugate model. The scale matrix parameter of the inverse-Wishart is set to the identity (default), whereas the degree of freedom parameter is estimated by maximization of the marginal likelihood. The function also computes the Bayes factor and tail probability (p-values) to test the marginal or conditional independence between all pairs of variables.

Usage

```
beam(X, type = "conditional", return.only = c("cor", "BF", "prob"),
verbose=TRUE, D=NULL)
```

Arguments

X	n by p data matrix
type	character. Either "marginal", "conditional" or "both". See Details.
return.only	character. Either "cor", "BF", "prob". See details.
verbose	logical. Whether information on progress should be be printed.
D	matrix. Prior marginal correlation matrix. Must be positive definite, well-conditioned and have unit variance.

Details

The arguments type and return.only have essentially been introduced for computational and memory savings. Using argument type the user may indicate whether the marginal dependencies ("marginal"), the conditional dependencies ("conditional") or both ("both") are to be inferred. On the other hand, the argument return.only is used to indicate whether the correlations ("cor"), Bayes factors ("BF") or tail probabilities ("prob") should be returned. Default is to return all three quantities for conditional dependencies.

Value

An object of class beam-class

Author(s)

Gwenael G.R. Leday and Ilaria Speranza

References

Leday, G.G.R. and Richardson, S. (2019). Fast Bayesian inference in large Gaussian graphical models. *Biometrics*.

Examples

```
# Load data
data(TCPAprad)

# beam
fit <- beam(X = TCPAprad, type="both")

# Print summary
summary(fit)

# Extract matrix of marginal correlations
mcor(fit)[1:5, 1:5]

# Extract matrix of partial correlations
pcor(fit)[1:5, 1:5]

# Plot log-marginal likelihood of the Gaussian conjugate model
plotML(fit)
```

```
# Plot heatmap of marginal (upper triangle) and/or
# partial (lower triangle) correlation estimates
plotCor(fit)
```

---

beam-class *Class beam*

Description

An S4 class representing the output of the beam function.

Usage

```
  ## S4 method for signature 'beam'
print(x, ...)

  ## S4 method for signature 'beam'
show(object)

  ## S4 method for signature 'beam'
summary(object, ...)

  ## S4 method for signature 'beam'
marg(object)

  ## S4 method for signature 'beam'
cond(object)

  ## S4 method for signature 'beam'
mcor(object)

  ## S4 method for signature 'beam'
pcor(object)

  ## S4 method for signature 'beam'
postExpSigma(object, vars.method="eb")

  ## S4 method for signature 'beam'
postExpOmega(object, vars.method="eb")

  ## S4 method for signature 'beam'
plotML(object, ...)

  ## S4 method for signature 'beam'
plotCor(object, type = object@type, order = 'original', by = "marginal")
```

```
  ## S4 method for signature 'beam'
bgraph(object)

  ## S4 method for signature 'beam'
ugraph(object)
```

Arguments

x	An object of class beam-class
object	An object of class beam-class
type	character. Type of correlation to be displayed (marginal, conditional or both)
order	character. Either 'original' or 'clust'. If 'clust' the rows and columns of the correlation matrix are reordered using the cluster memberships obtained by the Louvain clustering algorithm.
by	character. When type ="both" and order = 'clust', specifies whether the clustering has to be performed using the complete weighted marginal or conditional independence graph.
vars.method	character. Method of shrinkage estimation for the variances. Either 'eb', 'mean', 'median' for shrinkage estimation of variance respectively towards an estimated shrinkage target, the mean or the median of the sample variances. Choosing 'none' carries out no shrinkage and uses the sample variances, whereas choosing 'scaled' means that the sample covariance has unit diagonal.
...	further arguments passed to or from other methods.

Slots

table dat.frame. A data.frame containing marginal and/or partial correlation estimates, Bayes factors and tail probabilities for each edge.

deltaOpt numeric. Empirical Bayes estimate of hyperparameter delta.

alphaOpt numeric. Empirical Bayes estimate of hyperparameter alpha.

dimX numeric. Dimension of the input data matrix X.

type character. Input argument.)

varlabs character. Column labels of X.

gridAlpha matrix. A matrix containing the log-marginal likelihood of the Gaussian conjugate model as a function of a grid of values of alpha and delta.

valOpt numeric. Maximum value of the log-marginal likelihood of the Gaussian conjugate model.

return.only character. Input argument.

time numeric. Running time (in seconds).

TinvStdev numeric. Square root of partial variances.

s numeric. Sample variances.

rzij numeric. Statistics.

Author(s)

Gwenael G.R. Leday and Ilaria Speranza

`beam.select` *Edge selection with multiple testing and error control*

Description

Infer graphical structures by multiple testing

Usage

```
beam.select(object, thres = 0.1, method = "BH",
return.only = c(object@return.only, "adj"))
```

Arguments

`object`	An object of class `beam-class`.
`thres`	numeric. Threshold to be applied on adjusted tail probabilities.
`method`	character. Method to use for multiple comparison adjustment of tail probabilities.
`return.only`	character. Quantities to be returned.

Details

The argument `method` allows to adjust the tail probabilities obtained from the null distributions of the Bayes factors for multiple comparisons. Possible choices are: "holm", "bonferroni", "BH", "BY" and "HC". Apart from "HC", these are passed onto the R function `p.adjust` from package **stats** and we refer the user to its documentation for details. The method "HC" provides an optimal decision threshold based on the Higher Criticism score which is computed using the R function `hc.thresh` from package **fdrtool**. Again, we refer to the associated documentation for details.

The argument `return.only` allows to decide which quantities have to be in the output: it could be any subvector of c('cor', 'BF', 'prob', 'adj') (provided that the requested quantities have been computed in the beam object, except for adjusted probabilities). It can also be set to NULL: in this case, only the selected edges will be returned without any additional information. The default value for this argument are the columns present in the beam object plus the adjusted probabilities.

Value

An object of class `beam.select-class`

Author(s)

Gwenael G.R. Leday and Ilaria Speranza

References

Drton, M., & Perlman, M. D. (2007). Multiple testing and error control in Gaussian graphical model selection. Statistical Science, 430-449.
Goeman, J. J., & Solari, A. (2014). Multiple hypothesis testing in genomics. Statistics in medicine, 33(11), 1946-1978.
Donoho, D., & Jin, J. (2015). Higher criticism for large-scale inference, especially for rare and weak effects. Statistical Science, 30(1), 1-25.
Klaus, B., & Strimmer, K. (2012). Signal identification for rare and weak features: higher criticism or false discovery rates?. Biostatistics, 14(1), 129-143.

`beam.select-class` *Class beam.select*

Description

An S4 class representing the output of the `beam.select` function.

Usage

```
  ## S4 method for signature 'beam.select'
print(x, ...)

  ## S4 method for signature 'beam.select'
show(object)

  ## S4 method for signature 'beam.select'
summary(object, ...)

  ## S4 method for signature 'beam.select'
marg(object)

  ## S4 method for signature 'beam.select'
cond(object)

  ## S4 method for signature 'beam.select'
mcor(object)

  ## S4 method for signature 'beam.select'
pcor(object)

  ## S4 method for signature 'beam.select'
plotML(object, ...)

  ## S4 method for signature 'beam.select'
plotAdj(object, type=object@type, order = "original")

  ## S4 method for signature 'beam.select'
```

```
bgraph(object)

  ## S4 method for signature 'beam.select'
ugraph(object)
```

Arguments

x	An object of class beam.select-class
object	An object of class beam.select-class
type	character. Type of correlation to be displayed (marginal, conditional or both)
order	character. Either 'original' or 'clust'. If 'clust' the rows and columns of the adjacency matrix are reordered using the cluster memberships obtained by the Louvain clustering algorithm.
...	further arguments passed to or from other methods.

Slots

marginal data.frame. A data.frame containing the marginal correlation estimates, Bayes factors and tail probabilities for the selected edges only.

conditional data.frame. A data.frame containing the partial correlation estimates, Bayes factors and tail probabilities for the selected edges only.

dimX numeric. Dimension of the input data matrix X.

type character. Input type (marginal, conditional or both)

varlabs character. Column labels of X.

alphaOpt numeric. Empirical Bayes estimates of hyperparameter alpha.

gridAlpha matrix. A matrix containing the log-marginal likelihood of the Gaussian conjugate model as a function of a grid of values of alpha and delta.

valOpt numeric. Maximum value of the log-marginal likelihood of the Gaussian conjugate model

method character. Input method.

thres numeric. Input threshold

Author(s)

Gwenael G.R. Leday and Ilaria Speranza

`lightbeam` *Fast inference of a conditional independence graph*

Description

Fast and memory efficient reconstruction of large conditional independence networks.

Usage

```
lightbeam(X, thres = 0.1, method = "holm", verbose=TRUE)
```

Arguments

`X`	n by p data matrix
`thres`	numeric. Significance threshold to be applied on adjusted tail probabilities.
`method`	character. Method to use for multiple comparison adjustment of tail probabilities.
`verbose`	logical. Whether information on progress should be be printed.

Details

The function returns a sparse upper triangular matrix containing the (adjusted) tails probabilities that are below the given threshold (i.e. significant).

Value

An object of class `dgCMatrix-class`

Author(s)

Gwenael G.R. Leday

References

Leday, G.G.R. and Richardson, S. (2019). Fast Bayesian inference in large Gaussian graphical models. *Biometrics*.

Examples

```
# Load data
data(TCPAprad)

# beam
res <- lightbeam(X = TCPAprad, thres=0.1)
```

TCPAprad *Protein expression data.*

Description

Level 3 normalized proteomic data (v3.0) from The Cancer Proteome Alas (http://tcpaportal.org/tcpa). The data comprise the measurements of 189 antibodies obtained from 164 tumor tissue samples (prostate adenocarcinoma) using reverse phase protein arrays (RPPA).

Usage

```
TCPAprad
```

Format

A 164 by 189 matrix

Source

The Cancer Proteome Alas (http://tcpaportal.org/tcpa)

References

Li J, Lu Y, Akbani R, Ju Z, Roebuck PL, Liu W, Yang J-Y, Broom BM, Verhaak RGW, Kane DW, Wakefield C, Weinstein JN, Mills GB, Liang H. (2013). TCPA: A Resource for Cancer Functional Proteomics Data. *Nature Methods* 10(11), 1046-1047.

Examples

```
data(TCPAprad)
dim(TCPAprad)
TCPAprad[1:5, 1:5]
```

Index

beam, *2*, 3, *5*
beam-class, 5
beam-package, 2
beam.select, *3*, 7, *8*
beam.select-class, 8
bgraph *(*beam-class*)*, 5
bgraph,beam-method *(*beam-class*)*, 5
bgraph,beam.select-method
*(*beam.select-class*)*, 8

cond *(*beam-class*)*, 5
cond,beam-method *(*beam-class*)*, 5
cond,beam.select-method
*(*beam.select-class*)*, 8

lightbeam, 10

marg *(*beam-class*)*, 5
marg,beam-method *(*beam-class*)*, 5
marg,beam.select-method
*(*beam.select-class*)*, 8
mcor *(*beam-class*)*, 5
mcor,beam-method *(*beam-class*)*, 5
mcor,beam.select-method
*(*beam.select-class*)*, 8

pcor *(*beam-class*)*, 5
pcor,beam-method *(*beam-class*)*, 5
pcor,beam.select-method
*(*beam.select-class*)*, 8
plotAdj *(*beam.select-class*)*, 8
plotAdj,beam.select-method
*(*beam.select-class*)*, 8
plotCor *(*beam-class*)*, 5
plotCor,beam-method *(*beam-class*)*, 5
plotML *(*beam-class*)*, 5
plotML,beam-method *(*beam-class*)*, 5
plotML,beam.select-method
*(*beam.select-class*)*, 8
postExpOmega *(*beam-class*)*, 5
postExpOmega,beam-method *(*beam-class*)*, 5
postExpSigma *(*beam-class*)*, 5
postExpSigma,beam-method *(*beam-class*)*, 5
print,beam-method *(*beam-class*)*, 5
print,beam.select-method
*(*beam.select-class*)*, 8

show,beam-method *(*beam-class*)*, 5
show,beam.select-method
*(*beam.select-class*)*, 8
summary,beam-method *(*beam-class*)*, 5
summary,beam.select-method
*(*beam.select-class*)*, 8

TCPAprad, 11

ugraph *(*beam-class*)*, 5
ugraph,beam-method *(*beam-class*)*, 5
ugraph,beam.select-method
*(*beam.select-class*)*, 8