Package ‘equSA’

May 6, 2019

Type Package

Title Learning High-Dimensional Graphical Models

Version 1.2.1

Date 2019-05-04

Author Bochao Jia, Faming Liang, Runmin Shi, Suwa Xu

Maintainer Bochao Jia <jbc409@gmail.com>

Depends R (>= 3.0.2)

Imports igraph, huge, XMRF, ZIM, mvtnorm, speedglm, SIS, ncvreg, survival, bnlearn,doParallel, parallel, foreach

Description Provides an equivalent measure of partial correlation coefficients for high-dimensional Gaussian Graphical Models to learn and visualize the underlying relationships between variables from single or multiple datasets. You can refer to Liang, F., Song, Q. and Qiu, P. (2015) <doi:10.1080/01621459.2015.1012391> for more detail. Based on this method, the package also provides the method for constructing networks for Next Generation Sequencing Data, jointly estimating multiple Gaussian Graphical Models, constructing single graphical model for heterogeneous dataset, inferring graphical models from high-dimensional missing data and estimating moral graph for Bayesian network.

License GPL-2

LazyLoad true

NeedsCompilation yes

Repository CRAN

Date/Publication 2019-05-05 22:40:03 UTC

RoxygenNote 6.1.1

R topics documented:

`equSA-package` *Graphical model has been widely used in many scientific fileds to describe the conditional independent relationships for a large set of random variables. Through this package, we provide tools to learn structure for undirected graph (Markov Random Field) and moral graph for directed acyclic graph (Bayesian Network).*

Description

The package provides multiple algorithms for learning high-dimensional graphical models including both undirected graph and directed acyclic graph. For the undirect graph, the package provides an equivalent measure of partial correlation coefficients for high-dimensional Gaussian Graphical Models. Extended methods for inferring network structures from discretevariables are also available. Moreover, we also provide some methods for estimating graphical models from multiple datasets.

For the directed acyclic graph, the package provides the p-learning algorithm which is used to learn moral graphs in construction of high-dimensional Bayesian Networks for mixed data.

Details

Package:	equSA
Type:	Package
Version:	1.2.1
Date:	2019-05-04
License:	GPL-2

We propose an equvalent mearsure of partial correlation coeffient estimator called ψ estimators which enable us to estimate these networks via sparse, high-dimensional undirected graphical models. (Liang, F et al, 2015)

Here, we provide the community a convenient and useful tool to learn a Gaussian Graphical Models.

To estimate the network structures from Gaussian distributed data with this package, users simply need to specify the `"method"` in the main function, for example `equSAR(data,...)` to fit GGM to get the estimated adjacency matrix.

In this package, we also provide the code for combining Networks from two different dataset `combineR(data1,data2,...)` and the code for detecting difference between two Networks, for example `diffR(data1,data2,...)`. data1 and data2 should share the same dimension of variables (p) but allow have different samples (n).

Besides estimating single GGM, we also propose a joint estimation method for multiple GGMs. This is achieved by ψ- learning algorithm for graphical model at each time point combined with an Bayesian data integration method to estimate integrative ψ scores. Then multiple hypothesis tests were applied to identify the edges for each pair of variables. `JGGM(data,...)`.

If the data contains mixed types of variables, such as either Guassian or binary distributed. We provide a method for learning graphical models for this mixed dataset with edge restrictions option available, see `plearn.struct(data,...)`. We also provide a method for jointly estimation of mixed graphical model, see `JMGM(data,...)`.

If the data are not Gaussian distributed, for example, the count data, we propose a random effect model-based transformation to continuized data `ContTran(data,...)`, and then we transform the continuized data to Gaussian via a semiparametric transformation and then apply ψ- learning algorithm to reconstruct networks. The proposed method is consistent, and the resulting network satisfies the faithfulness and global Markov properties.The most common application is to estimate Gene Regulatory Networks from Next Generation Sequencing Data (Jia, B et al, 2017).

If we have the data following a distinct distribution and therefore produce the heterogeneous data. In this case, we might still be interested in constructing a single gene regulatory network for the heterogeneous data in a fashion of data integration, see `GGMM(data,...)`.

For learning high-dimensional Gaussian Graphical Models from missing data, we provide a Imputation-Regularized Optimzation (IRO) algorithm (Liang et al, 2018). See `GraphIRO(data,...)`.

For learning moral graph and markov blanket for Bayesian network, the package currently supports for Gaussian and binary data and also mixed type of data. See `plearn.moral(data,...)`. The proposed algorithm provides a feasible way to describe conditional dependence relationships for the directed acyclic graph.

To Construct confidence intervals and assess p-values in high-dimensional linear and generalized linear models. See `MNR(x,y,...)` for detail.

Author(s)

Bochao Jia, Faming Liang, Runmin Shi, Suwa Xu Maintainer: Bochao Jia<jbc409@gmail.com>

References

Friedman, J., Hastie, T., & Tibshirani, R. (2001). The elements of statistical learning (Vol. 1). Springer, Berlin: Springer series in statistics.

Liang, F., Song, Q. and Qiu, P. (2015). An Equivalent Measure of Partial Correlation Coefficients for High Dimensional Gaussian Graphical Models. J. Amer. Statist. Assoc., 110, 1248-1265.<doi:10.1080/01621459.2015.1012391>

Liang, F. and Zhang, J. (2008) Estimating FDR under general dependence using stochastic approximation. Biometrika, 95(4), 961-977.<doi:10.1093/biomet/asn036>

Liang, F., Jia, B., Xue, J., Li, Q., and Luo, Y. (2018). An Imputation Regularized Optimization Algorithm for High-Dimensional Missing Data Problems and Beyond. Submitted to Journal of the Royal Statistical Society Series B.

Liu, H., Lafferty, J. and Wasserman, L. (2009). The Nonparanormal: Semiparametric Estimation of High Dimensional Undirected Graphs. Journal of Machine Learning Research , 10, 2295-2328.

Jia, B., Xu, S., Xiao, G., Lamba, V., Liang, F. (2017) Inference of Genetic Networks from Next Generation Sequencing Data. Biometrics.

Jia, B., and Liang, F. (2018). A Fast Hybrid Bayesian Integrative Learning of Multiple Gene Regulatory Networks for Type 1 Diabetes. Submitted to Biostatistics.

Jia, B. and Liang, F. (2018). Learning Gene Regulatory Networks with High-Dimensional Heterogeneous Data. Accept by ICSA Springer Book.

Jean-Philippe, Pellet and Andre,Elisseeff (2008). Using Markov blankets for causal structure learning. Journal of Machine Learning Research, 9, 1295-1342.

Xu, S., Jia, B., and Liang, F. (2019). Learning Moral Graphs in Construction of High-Dimensional Bayesian Networks for Mixed Data. Neural computation, 1-32.

Jia, B., and Liang, F. (2018) Joint Estimation of Restricted Mixed Graphical Models. manuscript.

Liang, F., Xue, J. and Jia, B. (2018). Markov Neighborhood Regression for High-Dimensional Inference. Submitted to J. Amer. Statist. Assoc.

Examples

```
library(equSA)
data(TR0)
subset <- TR0
equSAR(subset)
```

alarm *One example dataset for p-plearning algorithm.*

Description

The ALARM ("A Logical Alarm Reduction Mechanism") is a Bayesian network designed to provide an alarm message system for patient monitoring.

alarm An object of class 'bn' for 'alarm' dataset. See package **bnlearn** for detail.

Usage

```
data(alarm)
```

Format

Alarm dataset is an object of class `bn` or `bn.fit`. See package **bnlearn** for detail.

References

Xu, S., Jia, B., and Liang, F. (2018). Learning Moral Graphs in Construction of High-Dimensional Bayesian Networks for Mixed Data. Submitted.

I. A. Beinlich, H. J. Suermondt, R. M. Chavez, and G. F. Cooper. The ALARM Monitoring System: A Case Study with Two Probabilistic Inference Techniques for Belief Networks. In Proceedings of the 2nd European Conference on Artificial Intelligence in Medicine, pages 247-256. Springer-Verlag, 1989.

combineR *Combine two networks.*

Description

Combine two networks to a single one.

Usage

```
combineR(Data1,Data2,ALPHA1=0.05,ALPHA2=0.05)
```

Arguments

`Data1` a n_1xp data matrix.

`Data2` a n_2xp data matrix.

`ALPHA1` The significance level of correlation screening for each dataset. In general, a high significance level of correlation screening will lead to a slightly large separator set S_{ij}, which reduces the risk of missing some important variables in the conditioning set. Including a few false variables in the conditioning set will not hurt much the accuracy of the ψ-partial correlation coefficient.

`ALPHA2` The significance level of ψ screening for integrative estimation of ψ scores.

Value

`A`	pxp Adjacency matrix of the combined graph.

Author(s)

Bochao Jia<`jbc409@gmail.com`> and Faming Liang

References

Liang, F., Song, Q. and Qiu, P. (2015). An Equivalent Measure of Partial Correlation Coefficients for High Dimensional Gaussian Graphical Models. J. Amer. Statist. Assoc., 110, 1248-1265.

Liang, F. and Zhang, J. (2008) Estimating FDR under general dependence using stochastic approximation. Biometrika, 95(4), 961-977.

Examples

```
library(equSA)
data(SR0)
data(TR0)
combineR(SR0,TR0)
```

`Cont2Gaus` *A transfomation from count data into Gaussian data*

Description

To transform count data into Gaussian distributed and also keep the consistency for contructing networks.

Usage

```
Cont2Gaus(iData,total_iteration=5000,stepsize=0.05)
```

Arguments

`iData`	a nxp count data matrix.
`total_iteration`	Total iteration number for Baysian random effect model-based transformation, default of 5000.
`stepsize`	The stepsize of updating parameters in transformation, default of 0.05.

Details

This is the function that transform the count data into Gaussian data which include two steps. First, we do data continuized transformation `ContTran(data,...)` and then we apply the semiparametric transformation (Liu, H et al, 2009) provided in **huge** packages to tranform continuized data into Gaussian distributed.

Value

`Gaus`	A nxp matrix of normalized data with Gaussian distribution.

Author(s)

Bochao Jia<`jbc409@gmail.com`> and Faming Liang

References

Jia, B., Xu, S., Xiao, G., Lamba, V., Liang, F. (2017) Inference of Genetic Networks from Next Generation Sequencing Data. Biometrics.

Liu, H., Lafferty, J. and Wasserman, L. (2009). The Nonparanormal: Semiparametric Estimation of High Dimensional Undirected Graphs. Journal of Machine Learning Research , 10, 2295-2328.

Examples

```
        library(equSA)
        data(count)
        Cont2Gaus(count,total_iteration=1000)
```

`ContSim` *A simulation method for generating count data from multivariate Zero-Inflated Negative Binomial distributions*

Description

Implements the data generation from multivariate Zero-Inflated Negative Binomial (ZINB) distributions with different graph structures, including `"random"`, `"hub"`, `"cluster"`, `"AR(2)"` and `"scale-free"`.

Usage

```
ContSim(n, p,  v = NULL, u = NULL, g = NULL, prob = NULL, vis = FALSE, verbose = TRUE,
graph.type="AR(2)", k=3.30, lambda=515, omega=0.003,lower.tail = TRUE, log.p = FALSE)
```

Arguments

`n` The number of observations (sample size).

`p` The number of variables (dimension).

`graph.type` The graph structure with 4 options: `"random"`, `"hub"`, `"cluster"`, `"AR(2)"` and `"scale-free"`.

`v` The off-diagonal elements of the precision matrix, controlling the magnitude of partial correlations with `u`. The default value is 0.3.

`u` A positive number being added to the diagonal elements of the precision matrix, to control the magnitude of partial correlations. The default value is 0.1.

`g` For `"cluster"` or `"hub"` graph, `g` is the number of hubs or clusters in the graph. The default value is about `d/20` if `d >= 40` and 2 if `d < 40`. NOT applicable to `"random"` and `"AR(2)"` graph.

`prob` For `"random"` graph, it is the probability that a pair of nodes has an edge. The default value is `3/d`. For `"cluster"` graph, it is the probability that a pair of nodes has an edge in each cluster. The default value is `6*g/d` if `d/g <= 30` and 0.3 if `d/g > 30`. NOT applicable to `"hub"` or `"AR(2)"` graphs.

`vis` Visualize the adjacency matrix of the true graph structure, the graph pattern, the covariance matrix and the empirical covariance matrix. The default value is `FALSE`

`verbose` If `verbose = FALSE`, tracing information printing is disabled. The default value is `TRUE`.

`k` dispersion parameter of ZINB distribution, default of 3.30.

`lambda` vector of (non-negative) means of ZINB distribution, default of 515.

`omega` zero-inflation parameter of ZINB distribution, default of 0.003.

`lower.tail` logical; if TRUE (default), probabilities are `P[X <= x]`, otherwise, `P[X> x]`.

`log.p` logical; if TRUE, probabilities `p` are given as `log(p)`.

Details

This is the function that can generate dataset from multivariate Zero-Inflated Negative Binomial distributions with different graph structures, including `"random"`, `"hub"`, `"cluster"`, `"AR(2)"` and `"scale-free"`.

Given the adjacency matrix `theta`, the graph patterns are generated as below:

(I) `random`: Each pair of off-diagonal elements are randomly set `theta[i,j]=theta[j,i]=1` for `i!=j` with probability `prob`, and `0` other wise. It results in about `d*(d-1)*prob/2` edges in the graph.

(II)`hub`:The row/columns are evenly partitioned into `g` disjoint groups. Each group is associated with a "center" row `i` in that group. Each pair of off-diagonal elements are set `theta[i,j]=theta[j,i]=1` for `i!=j` if `j` also belongs to the same group as `i` and `0` otherwise. It results in `d - g` edges in the graph.

(III)cluster:The row/columns are evenly partitioned into g disjoint groups. Each pair of off-diagonal elements are set theta[i,j]=theta[j,i]=1 for i!=j with the probability probif both i and j belong to the same group, and 0 other wise. It results in about g*(d/g)*(d/g-1)*prob/2 edges in the graph.

(IV)AR(2): The off-diagonal elements are set to be theta[i,j]=0.5 if |i-j|=1, theta[i,j]=0.05 if |i-j|=2 and 0 other wise.

(V) scale-free: The graph is generated using B-A algorithm. The initial graph has two connected nodes and each new node is connected to only one node in the existing graph with the probability proportional to the degree of the each node in the existing graph. It results in d edges in the graph.

The adjacency matrix theta has all diagonal elements equal to 0. To obtain a positive definite precision matrix, the smallest eigenvalue of theta*v (denoted by e) is computed. Then we set the precision matrix equal to theta*v+(|e|+0.1+u)I. The covariance matrix is then computed for generating multivariate ZINB dataset.

The default values for parameters k, lambda and omega of ZINB distribution are estimated from a real TCGA dataset. See Jia.B et al(2017) for more detail.

Value

A list of two elements:

data	The simulated count dataset in a nxp matrix.
Adj	pxp The adjacency matrix of true graph structure (in sparse matrix representation) for the generated data

Author(s)

Bochao Jia<jbc409@gmail.com>

References

Jia, B., Xu, S., Xiao, G., Lamba, V., Liang, F. (2017) Inference of Genetic Networks from Next Generation Sequencing Data. Biometrics.

T. Zhao and H. Liu.(2012) The huge Package for High-dimensional Undirected Graph Estimation in R. Journal of Machine Learning Research.

Yahav, I., and Shmueli, G. (2012). On generating multivariate Poisson data in management science applications. Applied Stochastic Models in Business and Industry, 28(1), 91-102.

Examples

```
library(equSA)
ContSim(100,200)
```

ContTran *A data continuized transformation*

Description

Transform count data into continuous data.

Usage

```
ContTran(iData,total_iteration=5000,stepsize=0.05)
```

Arguments

iData	a nxp count data matrix.
total_iteration	total iteration number for Baysian random effect model-based transformation, default of 5000.
stepsize	The stepsize of updating parameters in transformation, default of 0.05.

Details

This is the function that transform the count data into continuized data.

Value

continuz	nxp matrix of continuized data.

Author(s)

Bochao Jia<jbc409@gmail.com>, Suwa Xu and Faming Liang

References

Jia, B., Xu, S., Xiao, G., Lamba, V., Liang, F. (2017) Inference of Genetic Networks from Next Generation Sequencing Data. Biometrics.

Examples

```
library(equSA)
data(count)
ContTran(count,total_iteration=1000)
```

count *An example of count dataset for constructing network*

Description

A simulated dataset for illustrating our proposed method for inferening networks from next generation sequencing data.

Usage

```
data(count)
```

Format

count dataset is a 100x200 matrix. Each row represents a observation and each column represents a variable. It is generated from an overdispersion and zero-inflated Possion distribution.

References

Jia, B., Xu, S., Xiao, G., Lamba, V., Liang, F. (2017) Inference of Genetic Networks from Next Generation Sequencing Data. Biometrics.

DAGsim *Simulate a directed acyclic graph with mixed data (gaussian and binary)*

Description

Simulate a directed acyclic graph with mixed data (gaussian and binary).

Usage

```
DAGsim(n, p, sparsity = 0.02,  p.binary, type="AR(2)", verbose = TRUE)
```

Arguments

`n`	Number of observations.
`p`	Number of variables. Not applicable to the graph of `"alarm"` type.
`sparsity`	Sparsity of the graph in the `"random"` type, the default value is 0.02. Not applicable to other types.
`p.binary`	Number of binary variables. Not applicable to the graph of `"alarm"` type. The default value is `p/2`.
`type`	The graph structure with 3 options: `"random"`, `"alarm"` and `"AR(2)"` (default).
`verbose`	If `verbose = FALSE`, tracing information printing is disabled. The default value is `TRUE`.

Details

Given the type of graph, the patterns are generated as below:

(I) `"random"`: Each pair of off-diagonal elements are randomly set `edgematrix[i,j]=1` for `i < j` with probability `sparsity`, and `0` otherwise. It results in about `p*(p-1)*sparsity/2` edges in the graph.

(II)`"AR(2)"`: The off-diagonal elements are set to be `theta[i,j]=1` if `i<j` and `|i-j|<=2` and `0` otherwise.

(III) `"alarm"`: The graph structure is directly borrowed from package **'bnlearn'**, which has 37 variables with 46 edges. See **'bnlearn'** for more detail.

Value

A list of five objects.

`edgematrix` A pxp matrix which indicates the true structure of directed acyclic graph. If the (i,j)th element is equal to 1, there exists a directed edge from X_i to X_j.

`data` The simulated dataset in a nxp matrix.

`moral.matrix` The simulated adjacency matrix of the moral graph, which is the undircted version of Bayesian network.

`gaussian.index` The index of Gaussian variables.

`binary.index` The index of binary variables.

Author(s)

Suwa Xu, Bochao Jia and Faming Liang

References

Kalisch, M., and Buhlmann, P. (2007). Estimating high-dimensional directed acyclic graphs with the PC-algorithm. Journal of Machine Learning Research, 8(Mar), 613-636.

Xu, S., Jia, B., and Liang, F. (2018). Learning Moral Graphs in Construction of High-Dimensional Bayesian Networks for Mixed Data. Submitted.

I. A. Beinlich, H. J. Suermondt, R. M. Chavez, and G. F. Cooper. The ALARM Monitoring System: A Case Study with Two Probabilistic Inference Techniques for Belief Networks. In Proceedings of the 2nd European Conference on Artificial Intelligence in Medicine, pages 247-256. Springer-Verlag, 1989.

Examples

```
library(equSA)
DAGsim(n=300, p=100, type="AR(2)", p.binary=50)
```

diffR *Detect difference between two networks.*

Description

Detecting significant different edges between two networks.

Usage

```
diffR(Data1,Data2,ALPHA1=0.05,ALPHA2=0.05)
```

Arguments

Data1	a n_1xp data matrix.
Data2	a n_2xp data matrix.
ALPHA1	The significance level of correlation screening for each dataset. In general, a high significance level of correlation screening will lead to a slightly large separator set S_{ij}, which reduces the risk of missing some important variables in the conditioning set. Including a few false variables in the conditioning set will not hurt much the accuracy of the ψ-partial correlation coefficient.
ALPHA2	The significance level of ψ screening for integrative estimation of ψ scores.

Value

A	pxp adjacency matrix of the combined graph.

Author(s)

Bochao Jia<`jbc409@gmail.com`> and Faming Liang

References

Liang, F., Song, Q. and Qiu, P. (2015). An Equivalent Measure of Partial Correlation Coefficients for High Dimensional Gaussian Graphical Models. J. Amer. Statist. Assoc., 110, 1248-1265.

Liang, F. and Zhang, J. (2008) Estimating FDR under general dependence using stochastic approximation. Biometrika, 95(4), 961-977.

Examples

```
library(equSA)
data(SR0)
data(TR0)
diffR(SR0,TR0)
```

equSAR *An equvalent mearsure of partial correlation coeffients*

Description

Infer networks from Gaussian data by ψ-learning algorithm.

Usage

```
equSAR(iData,iMaxNei,ALPHA1=0.05,ALPHA2=0.05,GRID=2,iteration=100)
```

Arguments

iData	a nxp data matrix.
iMaxNei	Neiborhood size in correlation screening step, default to $n/log(n)$, where n is the number of observation.
ALPHA1	The significance level of correlation screening. In general, a high significance level of correlation screening will lead to a slightly large separator set S_{ij}, which reduces the risk of missing some important variables in the conditioning set. Including a few false variables in the conditioning set will not hurt much the accuracy of the ψ-partial correlation coefficient.
ALPHA2	The significance level of ψ screening.
GRID	The number of components for the ψ scores. The default value is 2.
iteration	Number of iterations for screening. The default value is 100.

Details

This is the main function of the package that fit the Gaussian Graphical Models and obtain the ψ scores and adjacency matrix.

Value

A list of two elements:

Adj	pxp adjacency matrix of the generated graph.
score	Estimated ψ score matrix which has 3 columns. The first two columns denote the pair indices of variables i and j and the last column denote the calculated ψ scores for this pair.

Author(s)

Bochao Jia and Faming Liang

References

Liang, F., Song, Q. and Qiu, P. (2015). An Equivalent Measure of Partial Correlation Coefficients for High Dimensional Gaussian Graphical Models. J. Amer. Statist. Assoc., 110, 1248-1265.

Liang, F. and Zhang, J. (2008) Estimating FDR under general dependence using stochastic approximation. Biometrika, 95(4), 961-977.

Examples

```
library(equSA)
data <- GauSim(100,100)$data
equSAR(data)
```

`GauSim` *Simulate centered Gaussian data from multiple types of structures.*

Description

Implements the data generation from Gaussian distribution with different graph structures, including `"random"`, `"hub"`, `"cluster"`, `"AR(2)"` and `"scale-free"`.

Usage

```
GauSim(n, p, graph = "AR(2)", v = NULL, u = NULL, g = NULL, prob = NULL,
vis = FALSE, verbose = TRUE)
```

Arguments

`n`	The number of observations (sample size).
`p`	The number of variables (dimension).
`graph`	The graph structure with 4 options: `"random"`, `"hub"`, `"cluster"`, `"AR(2)"` and `"scale-free"`.
`v`	The off-diagonal elements of the precision matrix, controlling the magnitude of partial correlations with `u`. The default value is `0.3`.
`u`	A positive number being added to the diagonal elements of the precision matrix, to control the magnitude of partial correlations. The default value is `0.1`.
`g`	For `"cluster"` or `"hub"` graph, `g` is the number of hubs or clusters in the graph. The default value is about `d/20` if `d >= 40` and `2` if `d < 40`. NOT applicable to `"random"` and `"AR(2)"` graph.
`prob`	For `"random"` graph, it is the probability that a pair of nodes has an edge. The default value is `3/d`. For `"cluster"` graph, it is the probability that a pair of nodes has an edge in each cluster. The default value is `6*g/d` if `d/g <= 30` and `0.3` if `d/g > 30`. NOT applicable to `"hub"` or `"AR(2)"` graphs.

`vis` Visualize the adjacency matrix of the true graph structure, the graph pattern, the covariance matrix and the empirical covariance matrix. The default value is `FALSE`

`verbose` If `verbose = FALSE`, tracing information printing is disabled. The default value is `TRUE`.

Details

Given the adjacency matrix `theta`, the graph patterns are generated as below:

(I) `random`: Each pair of off-diagonal elements are randomly set `theta[i,j]=theta[j,i]=1` for `i!=j` with probability `prob`, and `0` other wise. It results in about `d*(d-1)*prob/2` edges in the graph.

(II)`hub`:The row/columns are evenly partitioned into `g` disjoint groups. Each group is associated with a "center" row `i` in that group. Each pair of off-diagonal elements are set `theta[i,j]=theta[j,i]=1` for `i!=j` if `j` also belongs to the same group as `i` and `0` otherwise. It results in `d - g` edges in the graph.

(III)`cluster`:The row/columns are evenly partitioned into `g` disjoint groups. Each pair of off-diagonal elements are set `theta[i,j]=theta[j,i]=1` for `i!=j` with the probability `probif` both `i` and `j` belong to the same group, and `0` other wise. It results in about `g*(d/g)*(d/g-1)*prob/2` edges in the graph.

(IV)`AR(2)`: The off-diagonal elements are set to be `theta[i,j]=0.5` if `|i-j|=1`, `theta[i,j]=0.05` if `|i-j|=2` and `0` other wise.

(V) `scale-free`: The graph is generated using B-A algorithm. The initial graph has two connected nodes and each new node is connected to only one node in the existing graph with the probability proportional to the degree of the each node in the existing graph. It results in `d` edges in the graph.

The adjacency matrix `theta` has all diagonal elements equal to `0`. To obtain a positive definite precision matrix, the smallest eigenvalue of `theta*v` (denoted by `e`) is computed. Then we set the precision matrix equal to `theta*v+(|e|+0.1+u)I`. The covariance matrix is then computed to generate multivariate normal data.

Value

A list of three elements:

`data` The simulated Gaussian distributed dataset with mean 0 in a nxp matrix.

`sigma` pxp The The covariance matrix for the generated data.

`theta` pxp The adjacency matrix of true graph structure (in sparse matrix representation) for the generated data.

Author(s)

Bochao Jia<`jbc409@gmail.com`>

References

Jia, B., Xu, S., Xiao, G., Lamba, V., Liang, F. (2017) Inference of Genetic Networks from Next Generation Sequencing Data. Biometrics.

T. Zhao and H. Liu.(2012) The huge Package for High-dimensional Undirected Graph Estimation in R. Journal of Machine Learning Research.

Examples

```
library(equSA)
GauSim(100,200)
```

GGMM *Learning high-dimensional Gaussian Graphical Models with Heterogeneous Data.*

Description

Gaussian Graphical Mixture Models for learning a single high-dimensional network structure from heterogeneous dataset.

Usage

```
GGMM(data, A, M, alpha1 = 0.1, alpha2 = 0.05, alpha3 = 0.05, iteration = 30, warm = 20)
```

Arguments

`data` nxp mixture Gaussian distributed dataset.

`A` pxp true adjacency matrix for evaluating the performance.

`M` The number of heterogeneous groups.

`alpha1` The significance level of correlation screening in the ψ-learning algorithm, see R package **equSA** for detail. In general, a high significance level of correlation screening will lead to a slightly large separator set, which reduces the risk of missing important variables in the conditioning set. In general, including a few false variables in the conditioning set will not hurt much the accuracy of the ψ-partial correlation coefficient, the default value is 0.1.

`alpha2` The significance level of ψ-partial correlation coefficient screening for estimating the adjacency matrix, see **equSA**, the default value is 0.05.

`alpha3` The significance level of integrative ψ-partial correlation coefficient screening for estimating the adjacency matrix of GGMM method, the default value is 0.05.

`iteration` The number of total iterations, the default value is 30.

`warm` The number of burn-in iterations, the default value is 20.

Value

`RecPre` The output of Recall and Precision values of our proposed method.

`Adj` pxp Estimated adjacency matrix.

`label` The estimated group indices for each observation.

`BIC` The BIC scores for determining the number of groups M.

Author(s)

Bochao Jia<`jbc409@gmail.com`> and Faming Liang

References

Liang, F., Song, Q. and Qiu, P. (2015). An Equivalent Measure of Partial Correlation Coefficients for High Dimensional Gaussian Graphical Models. J. Amer. Statist. Assoc., 110, 1248-1265.

Liang, F. and Zhang, J. (2008) Estimating FDR under general dependence using stochastic approximation. Biometrika, 95(4), 961-977.

Liang, F., Jia, B., Xue, J., Li, Q., and Luo, Y. (2018). An Imputation Regularized Optimization Algorithm for High-Dimensional Missing Data Problems and Beyond. Submitted to Journal of the Royal Statistical Society Series B.

Jia, B. and Liang, F. (2018). Learning Gene Regulatory Networks with High-Dimensional Heterogeneous Data. Accept by ICSA Springer Book.

Examples

```
library(equSA)
result <- SimHetDat(n = 100, p = 200, M = 3, mu = 0.5, type = "band")
Est <- GGMM(result$data, result$A, M = 3, iteration = 30, warm = 20)
## plot network by our estimated adjacency matrix.
plotGraph(Est$Adj)
## plot the Recall-Precision curve
plot(Est$RecPre[,1], Est$RecPre[,2], type="l", xlab="Recall", ylab="Precision")
```

`GraphIRO` *Learning high-dimensional Gaussian Graphical Models with Missing Observations.*

Description

The imputation regularized optimization (IRO) algorithm for learning high-dimensional Gaussian Graphical Models from incomplete dataset.

Usage

```
GraphIRO(data, A, alpha1 = 0.05, alpha2 = 0.05, alpha3 = 0.05, iteration = 30, warm = 10)
```

Arguments

`data` nxp Dataset with missing values.

`A` True adjacency matrix for evaluating the performance of the IRO algorithm.

`alpha1` The significance level of correlation screening in the ψ-learning algorithm, see R package **equSA** for detail. In general, a high significance level of correlation screening will lead to a slightly large separator set, which reduces the risk of missing important variables in the conditioning set. In general, including a few false variables in the conditioning set will not hurt much the accuracy of the ψ-partial correlation coefficient, the default value is 0.05.

`alpha2` The significance level of ψ-partial correlation coefficient screening for estimating the adjacency matrix, see **equSA**, the default value is 0.05.

`alpha3` The significance level of integrative ψ-partial correlation coefficient screening for estimating the adjacency matrix of IRO_Ave method, the default value is 0.05.

`iteration` The number of total iterations, the default value is 30.

`warm` The number of burn-in iterations, the default value is 10.

Value

`RecPre` The output of Recall and Precision values for the IRO algorithm.

`Adj` pxp Estimated adjacency matrix by our IRO algorithm.

Author(s)

Bochao Jia<`jbc409@gmail.com`> and Faming Liang

References

Liang, F., Song, Q. and Qiu, P. (2015). An Equivalent Measure of Partial Correlation Coefficients for High Dimensional Gaussian Graphical Models. J. Amer. Statist. Assoc., 110, 1248-1265.

Liang, F. and Zhang, J. (2008) Estimating FDR under general dependence using stochastic approximation. Biometrika, 95(4), 961-977.

Liang, F., Jia, B., Xue, J., Li, Q., and Luo, Y. (2018). An Imputation Regularized Optimization Algorithm for High-Dimensional Missing Data Problems and Beyond. Submitted to Journal of the Royal Statistical Society Series B.

Examples

```
library(equSA)
result <- SimGraDat(n = 200, p = 100, type = "band", rate = 0.1)
Est <- GraphIRO(result$data, result$A, iteration = 20, warm = 10)
```

```
## plot network by our estimated adjacency matrix.
plotGraph(Est$Adj)
## plot the Recall-Precision curve.
plot(Est$RecPre[,1], Est$RecPre[,2], type="l", xlab="Recall", ylab="Precision")
```

`JGGM` *Joint estimation of Multiple Gaussian Graphical Models*

Description

Infer networks from Multiple Gaussian data from differnt groups using our proposed fast Bayesian integrative method.

Usage

```
JGGM(data,ALPHA1=0.05,ALPHA2=0.01,structure = "temporal",parallel=FALSE,nCPUs)
```

Arguments

`data` a list of nxp data matrices. n can be different for each dataset but p should be the same.

`ALPHA1` The significance level of correlation screening. In general, a high significance level of correlation screening will lead to a slightly large separator set S_{ij}, which reduces the risk of missing some important variables in the conditioning set. Including a few false variables in the conditioning set will not hurt much the accuracy of the ψ-partial correlation coefficient.

`ALPHA2` The significance level of ψ screening.

`structure` The depedent structure of multiple networks, either "temporal" or "spatial". The default is "temporal".

`parallel` Should parallelization be used? (logical), default is `FALSE`.

`nCPUs` Number of cores used for parallelization. Recommend to be equal to the number of datasets.

Details

This is the function that can jointly estimate multiple GGMs which can integrate the information throughtout all datasets. The method mainly consists three steps: (i) separate estimation of ψ-scores for each dataset, (ii) identifies possible changes of each edge across different groups and integrate the ψ scores across different groups simultaneously and (iii) apply multiple hypothesis test to identify edges using integrated ψ scores. See Jia, B., et al (2018).

Value

A list of three elements:

`A` An array of multiple adjacency matrices of networks which is a Mxpxp array. M is the number of dataset groups, p is the dimension of variables in each group.

`score.sep` Separately estimated ψ scores matrix for all pairs in multiple datasets. The first two columns denote the pair indices of variables i and j and the rest columns denote the estimated ψ scores for this pair in different groups.

`score.joint` Estimated integrative ψ scores matrix for all pairs in multiple datasets. The first two columns denote the pair indices of variables i and j and the rest columns denote the estimated integrative ψ scores for this pair in different groups.

Author(s)

Bochao Jia<`jbc409@gmail.com`> and Faming Liang

References

Jia, B., and Liang, F. (2018). A Fast Hybrid Bayesian Integrative Learning of Multiple Gene Regulatory Networks for Type 1 Diabetes. Submitted to Biostatistics.

Examples

```
library(equSA)
data(SR0)
data(TR0)
data_all <- vector(”list”,2)
data_all[[1]] <- SR0
data_all[[2]] <- TR0
JGGM(data_all,ALPHA1=0.05,ALPHA2=0.05)
```

`JMGM` *Joint Mixed Graphical Models*

Description

Infer network structures from multiple datasets with mixed types of variables and edge restrictions option available.

Usage

```
JMGM(data,ALPHA1=0.05,ALPHA2=0.01,restrict=FALSE,parallel=FALSE,nCPUs)
```

Arguments

`data` a list of nxp data matrices. n can be different for each dataset but p should be the same.

`ALPHA1` The significance level of correlation screening. In general, a high significance level of correlation screening will lead to a slightly large separator set S_{ij}, which reduces the risk of missing some important variables in the conditioning set. Including a few false variables in the conditioning set will not hurt much the accuracy of the ψ-partial correlation coefficient.

`ALPHA2` The significance level of ψ screening.

`restrict` Should edge restriction applied? (logical). If `TRUE`, we assume that there should be no edge among binary variables. The default is `FALSE`.

`parallel` Should parallelization be used? (logical), default is `FALSE`.

`nCPUs` Number of cores used for parallelization. Recommend to be equal to the number of datasets.

Details

This is the function that can jointly estimate multiple graphical models with mixed types of data and also consider the edge restriction scenarios. The method has three novelties: First, the proposed method resolves the conditional independence information using a p-learning algorithm and therefore can be applied to the mixed types of random variables. Second, the proposed method can construct networks with restricted edges determined by some preliminary knowledges. Third, the proposed method involves a Fast Bayesian joint estimation method which works on edge-wise scores and can achieve both fast and accurate integration performance for constructing multiple networks. See Jia and Liang (2018).

Value

A list of three elements:

`A` An array of multiple adjacency matrices of networks which is a Mxpxp array. M is the number of dataset groups, p is the dimension of variables in each group.

`score.sep` Separately estimated ψ scores matrix for all pairs in multiple datasets. The first two columns denote the pair indices of variables i and j and the rest columns denote the estimated ψ scores for this pair in different groups.

`score.joint` Estimated integrative ψ scores matrix for all pairs in multiple datasets. The first two columns denote the pair indices of variables i and j and the rest columns denote the estimated integrative ψ scores for this pair in different groups.

Author(s)

Bochao Jia<`jbc409@gmail.com`> and Faming Liang

References

Jia, B., and Liang, F. (2018) Joint Estimation of Restricted Mixed Graphical Models. manuscript.

Examples

```
library(equSA)
data1 <- DAGsim(n=200, p=100, type=”AR(2)”)$data
data2 <- DAGsim(n=200, p=100, type=”AR(2)”)$data
data_all <- vector(”list”,2)
data_all[[1]] <- data1
data_all[[2]] <- data2
JMGM(data_all,ALPHA1=0.1,ALPHA2=0.05,parallel=TRUE,nCPUs=2)
```

MNR *Markov Neighborhood Regression for High-Dimensional Inference.*

Description

Construct confidence intervals and assess p-values in high-dimensional linear and generalized linear models.

Usage

```
MNR(x,y,family='gaussian',penalty='lasso',tune='bic',alpha1=0.1,alpha2=0.05,level=0.95)
```

Arguments

`x` The design matrix, of dimensions nxp, without an intercept. Each row is an observation vector.

`y` The response vector of dimension nx1. Quantitative for family=’gaussian’, binary (0-1) for family=’binomial’. For family=’cox’, y should be an object of class `Surv`, as provided by the function `Surv()` in the package **survival**.

`family` Response type (see above).

`penalty` The penalty to be applied in the regularized likelihood subproblems. ’lasso’ (the default), ’MCP’, or ’SCAD’ are provided. See package **SIS** for detail.

`tune` Method for tuning the regularization parameter of the penalized likelihood subproblems and of the final model selected by (I)SIS. Options include tune=’bic’, tune=’ebic’, tune=’aic’, and tune=’cv’.

`alpha1` The significance level of correlation screening in the ψ-learning algorithm, see R package **equSA** for detail. In general, a high significance level of correlation screening will lead to a slightly large separator set, which reduces the risk of missing important variables in the conditioning set. In general, including a few false variables in the conditioning set will not hurt much the accuracy of the ψ-partial correlation coefficient, the default value is 0.1.

`alpha2` The significance level of ψ-partial correlation coefficient screening for estimating the adjacency matrix, see **equSA**, the default value is 0.05.

`level` the confidence level required, the default value is 0.95

Value

`CI` Estimated confidence intervals for all coefficients.

`coef` px1 estimated regression coefficients for all variables.

`pvalue` px1 estimated p-values for all variables.

Author(s)

Bochao Jia<`jbc409@gmail.com`> and Faming Liang

References

Liang, F., Xue, J. and Jia, B. (2018). Markov Neighborhood Regression for High-Dimensional Inference. Submitted to J. Amer. Statist. Assoc.

Examples

```
library(equSA)
p <- 500
coef_true <- rep(0,p)
coef_true[1:5] <- c(2,4,-3,-5,10)
coef <- c(1,coef_true)
data <- SimMNR(n = 200, p = 500, coef = coef, family = "gaussian")
MNR(data$x, data$y, family = "gaussian")
```

`Mulpval` *Multiple hypothesis tests for p values*

Description

Conduct multiple hypothesis tests from p values.

Usage

```
Mulpval(pvalue, ALPHA2=0.05,GRID=2,iteration=100)
```

Arguments

pvalue	A vector of p values.
ALPHA2	The significance level of screening, default of 0.05.
GRID	The number of components for the z-scores. The default value is 2.
iteration	Number of iterations for screening. The default value is 100.

Details

This is the function that conduct multiple hypothesis test for p values.

Value

qqqscore	The threshold of p value which indicates that p values are not larger than the threshold are considered significance and larger otherwise.

Author(s)

Bochao Jia, Faming liang<fmliang@purdue.edu>

References

Liang, F. and Zhang, J. (2008) Estimating FDR under general dependence using stochastic approximation. Biometrika, 95(4), 961-977.

Examples

```
library(equSA)
pvalue <- c(runif(20,0,0.001),runif(200,0,1))
Mulpval(pvalue,ALPHA2=0.05)
```

---

pcorselR *Multiple hypothesis test*

Description

Infer networks from ψ scores using multiple hypothesis test in ψ screening procedure.

Usage

```
pcorselR(score, ALPHA2=0.05,GRID=2,iteration=100)
```

Arguments

score	ψ score matrix which has 3 columns. The first two columns denote the pair of variables i and j and the last column denote the calculated ψ scores for this pair.
ALPHA2	The significance level of ψ screening, default of 0.05.
GRID	The number of components for the ψ-scores. The default value is 2.
iteration	Number of iterations for screening. The default value is 100.

Details

This is the function that conduct multiple hypothesis test for ψ scores, thus we called it ψ screening procedure.

Value

qqqscore	The threshold value of ψ scores which indicates that if one pair of variables has larger ψ scores than this threshold value in the ψ score matrix, this pair is considered as connected, i.e there is an edge between this pair of variables.

Author(s)

Bochao Jia, Faming liang<fmliang@purdue.edu>

References

Liang, F. and Zhang, J. (2008) Estimating FDR under general dependence using stochastic approximation. Biometrika, 95(4), 961-977.

Examples

```
library(equSA)
data(SR0)
U <- psical(SR0, ALPHA1=0.05,iteration=50)
##   probit transformation for psi scores ###
z<-U[,3]
q<-pnorm(-abs(z), log.p=TRUE)
q<-q+log(2.0)
s<-qnorm(q,log.p=TRUE)
s<-(-1)*s
U<-cbind(U[,1:2],s)
## subsampling for psi scores ###
N <- length(U[,1])
ratio<-ceiling(N/100000)
U<-U[order(U[,3]), 1:3]
m<-floor(N/ratio)
m0<-N-m*ratio
s<-sample.int(ratio,m,replace=TRUE)
for(i in 1:length(s)) s[i]<-s[i]+(i-1)*ratio
if(m0>0){
  s0<-sample.int(m0,1)+length(s)*ratio
  s<-c(s,s0)
}
Us<-U[s,]
y <- round(Us,6)
##  multiple hypothesis tests ###
pcorselR(y,ALPHA2=0.05)
```

plearn.moral *Learning Moral graph based on p-learning algorithm.*

Description

Construct moral graph of Bayeisan network for mixed types of random varaibles based on p-learning algorithm. Each variable in the dataset can be either binary or Gaussian distributed.

Usage

```
plearn.moral(data, gaussian.index = NULL, binary.index = NULL,
alpha1 = 0.1, alpha2 = 0.02, restrict = FALSE, score.only=FALSE)
```

Arguments

`data`	The data matrix, of dimensions nxp. Each row is an observation vector and each column is a variable.
`gaussian.index`	The index vector of Gaussian nodes. The default value is `NULL`. If not specified, the system will automatically determine the index for each variable.
`binary.index`	The index vector of binary nodes. The default value is `NULL`. If not specified, the system will automatically determine the index for each variable.
`alpha1`	The significant level of correlation screening in p-learning algorithm. The default value is 0.1.
`alpha2`	The significant level of partial correlation screening in p-learning algorithm. The dafault value is 0.02.
`restrict`	Should edge restriction applied? (logical). If `TRUE`, we assume that there should be no edge among binary variables. The default is `FALSE`.
`score.only`	If `TRUE`, it only reports z-scores for all pair of variables. The default is `FALSE`.

Details

This is the function that implements the p-learning algorithm for learning moral graph of Bayesian Network with mixed type of random variables.

Value

A list of two objects.

`moral.matrix`	The estimated adjacency matrix of moral graph.
`score`	The estimated z-scores for all pair of variables.

Author(s)

Suwa Xu Bochao Jia and Faming Liang

References

Xu, S., Jia, B., and Liang, F. (2018). Learning Moral Graphs in Construction of High-Dimensional Bayesian Networks for Mixed Data. Submitted.

Examples

```
library(equSA)
data.graph <- DAGsim(n = 200, p = 100, type="AR(2)", p.binary = 50)$data
plearn.moral(data.graph, alpha1 = 0.1, alpha2 = 0.02)
```

---

plearn.struct *Infer network structure for mixed types of random variables.*

Description

Learning graphical model structure for mixed types of random varaibles based on p-learning algorithm. Each variable in the dataset can be either binary or Gaussian distributed.

Usage

```
plearn.struct(data, gaussian.index = NULL, binary.index = NULL,
alpha1 = 0.1, alpha2 = 0.02, restrict = FALSE, score.only=FALSE)
```

Arguments

`data`	The data matrix, of dimensions nxp. Each row is an observation vector and each column is a variable.
`gaussian.index`	The index vector of Gaussian nodes. The default value is `NULL`. If not specified, the system will automatically determine the index for each variable.
`binary.index`	The index vector of binary nodes. The default value is `NULL`. If not specified, the system will automatically determine the index for each variable.
`alpha1`	The significant level of parent and children screening in p-learning algorithm. The default value is 0.1.
`alpha2`	The significant level of moral graph screening in p-learning algorithm. The dafault value is 0.02.

`restrict` Should edge restriction applied? (logical). If `TRUE`, we assume that there should be no edge among binary variables. The default is `FALSE`.

`score.only` If `TRUE`, it only reports z-scores for all pair of variables. The default is `FALSE`.

Details

This is the function that implements the p-learning algorithm for learning the undirect network structure for mixed types of random variables.

Value

A list of two objects.

`Adj` The estimated adjacency matrix of undirect network.

`score` The estimated z-scores for all pair of variables.

Author(s)

Bochao Jia and Faming Liang

References

Jia, B., and Liang, F. (2018) Joint Estimation of Restricted Mixed Graphical Models. manuscript.

Examples

```
library(equSA)
data.graph <- DAGsim(n = 200, p = 100, type="AR(2)", p.binary = 50)$data
plearn.struct(data.graph, alpha1 = 0.1, alpha2 = 0.02)
```

plotGraph *Plot Single Network*

Description

Plot a network with specific layout.

Usage

```
plotGraph(net, fn = "", th = 1e-06, mylayout = NULL)
```

Arguments

`net` a square adjacency matrix of the network to be plotted.

`fn` file name to save the network plot. Default to be an empty string, so the network is plotted to the standard output (screen). NOTE: if a file name is specified, it should be file name for PDF file.

`th` numeric value, default to 1e-06. To specify the threshold if the estimated coefficient between two variables is to be considered connected.

`mylayout` graph layout to draw the network, default to `NULL`.

Details

This function serves as the alternative plotting function to allow users to plot a specific network with specific layout, such as plotting the simulated network.

Value

Returns the layout object from **igraph** package - numeric matrix of two columns and the rows with the same number as the number of vertices.

Examples

```
library(equSA)
Adj <- GauSim(100,200,graph="scale-free")$theta
plotGraph(Adj)
```

plotJGraph *Plot Networks*

Description

Plot multiple networks with specific layout.

Usage

```
plotJGraph(A,fn="Net",th = 1e-06, mylayout = NULL)
```

Arguments

`A` An array of multiple adjacency matrices of networks to be plotted which is a Mxpxp array. M is the number of dataset groups, p is the dimension of variables in each group.

`fn` file name to save the network plots. Default to be an string called "Net". NOTE: It should be file name for PDF file.

`th` numeric value, default to 1e-06. To specify the threshold if the estimated coefficient between two variables is to be considered connected.

`mylayout` graph layout to draw networks, default to `NULL`.

Details

This function serves as the alternative plotting function to allow users to plot multiple networks with specific layout, such as plotting the simulated networks.

Value

Returns the multiple layout objects from **igraph** package - numeric matrix of two columns and the rows with the same number as the number of vertices.

Author(s)

Bochao Jia<jbc409@gmail.com>, Faming liang

References

Jia, B., and Liang, F. (2018). Learning Multiple Gene Regulatory Networks in Type 1 Diabetes through a Fast Bayesian Integrative Method. Submitted to Journal of Statistical Computing.

Examples

```
library(equSA)
data(SR0)
data(TR0)
data_all <- vector("list",2)
data_all[[1]] <- SR0
data_all[[2]] <- TR0
A <- JGGM(data_all,ALPHA1=0.05,ALPHA2=0.01)$Array
plotJGraph(A)
```

---

psical *A calculation of ψ scores.*

Description

To compute an equvalent mearsure of partial correlation coeffients called ψ scores.

Usage

```
psical(iData,iMaxNei,ALPHA1=0.05,GRID=2,iteration=100)
```

Arguments

`iData` a nxp data matrix.

`iMaxNei` Neiborhood size in correlation screening step, default to $n/log(n)$, where n is the number of observation.

`ALPHA1` The significance level of correlation screening. In general, a high significance level of correlation screening will lead to a slightly large separator set S_{ij}, which reduces the risk of missing some important variables in the conditioning set. Including a few false variables in the conditioning set will not hurt much the accuracy of the ψ-partial correlation coefficient.

`GRID` The number of components for the corrlation scores. The default value is 2.

`iteration` Number of iterations for screening. The default value is 100.

Details

This is the function to calculate ψ scores and can be used in combining or detecting difference of two networks.

Value

`score` Estimated ψ score matrix which has 3 columns. The first two columns denote the pair indices of variables i and j and the last column denote the calculated ψ scores for this pair.

Author(s)

Bochao Jia, Faming liang<`fmliang@purdue.edu`>

References

Liang, F., Song, Q. and Qiu, P. (2015). An Equivalent Measure of Partial Correlation Coefficients for High Dimensional Gaussian Graphical Models. J. Amer. Statist. Assoc., 110, 1248-1265.

Liang, F. and Zhang, J. (2008) Estimating FDR under general dependence using stochastic approximation. Biometrika, 95(4), 961-977.

Examples

```
library(equSA)
data <- GauSim(100,100)$data
psical(data)
```

SimGraDat — *Simulate Incomplete Data for Gaussian Graphical Models*

Description

Simulate compeletely missing at random (CMAR) data with a band structure, which can be used in `GraphIRO(data,...)` for estimating the structure of the Gaussian graphical network.

Usage

```
SimGraDat(n = 200, p = 100, type = "band", rate = 0.1)
```

Arguments

`n`	Number of observations, default of 200.
`p`	Number of covariates, default of 100.
`type`	`type=="band"` which denotes the band structure, with precision matrix $$C_{i,j} = \begin{cases} 0.5, & \text{if } \lvert j-i\rvert = 1, i = 2, ..., (p-1), \\ 0.25, & \text{if } \lvert j-i\rvert = 2, i = 3, ..., (p-2), \\ 1, & \text{if } i = j, i = 1, ..., p, \\ 0, & \text{otherwise.} \end{cases}$$
`rate`	Missing rate, the default value is 0.1.

Value

`data`	nxp Gaussian distributed data with missing.
`A`	pxp adjacency matrix used for generating data.

Author(s)

Bochao Jia<`jbc409@gmail.com`> and Faming Liang

References

Liang, F., Jia, B., Xue, J., Li, Q., and Luo, Y. (2018). An Imputation Regularized Optimization Algorithm for High-Dimensional Missing Data Problems and Beyond. Submitted to Journal of the Royal Statistical Society Series B.

Examples

```
library(equSA)
SimGraDat(n = 200, p = 100, type = "band", rate = 0.1)
```

SimHetDat *Simulate Heterogeneous Data for Gaussian Graphical Models*

Description

Simulate Heterogeneous data with a band structure, which can be used in `GGMM(data,...)` for estimating the structure of the Gaussian graphical network.

Usage

```
SimHetDat(n = 100, p = 200, M = 3, mu = 0.3, type = "band")
```

Arguments

`n`	Number of observations for each group, default of 100.
`p`	Number of covariates for each observation, default of 200.
`M`	Number of latent groups for the simulated dataset choose 2 or 3, default of 3.
`mu`	The mean difference among groups. If $M = 3$, the mean of three groups are $-mu, 0, mu$, respectively. If $M = 2$, the mean of two groups are $0, mu$, respectively.
`type`	`type=="band"` which denotes the band structure, with precision matrix

$$C_{i,j} = \begin{cases} 0.5, & \text{if } |j-i| = 1, i = 2, ..., (p-1), \\ 0.25, & \text{if } |j-i| = 2, i = 3, ..., (p-2), \\ 1, & \text{if } i = j, i = 1, ..., p, \\ 0, & \text{otherwise.} \end{cases}$$

Value

`data`	nxp Heterogeneous Gaussian distributed data.
`A`	pxp adjacency matrix used for generating data.
`label`	The group indices for each observation.

Author(s)

Bochao Jia<`jbc409@gmail.com`> and Faming Liang

References

Jia, B. and Liang, F. (2018). Learning Gene Regulatory Networks with High-Dimensional Heterogeneous Data. Accept by ICSA Springer Book.

Examples

```
library(equSA)
SimHetDat(n = 100, p = 200, M = 3, mu = 0.5, type = "band")
```

SimMNR *Simulate Data for high-dimensional inference*

Description

Simulate data with graphical structure for generalized regression, which can be used in `MNR(x,y,...)` for constructing confidence intervals and assessing p-values.

Usage

```
SimMNR(n, p, coef, family="gaussian")
```

Arguments

`n`	Number of observations.
`p`	Number of variables.
`coef`	A $p + 1$x1 vector. The first value denotes the intercept term and other p values denote the true regression coefficients for p variables.
`family`	Quantitative for family='gaussian' (default), binary (0-1) for family='binomial'. Survival data for family='cox'.

Details

We generate p variables from the following precision matrix, which is often been called "band" structure or "AR(2)" structure.

$$C_{i,j} = \begin{cases} 0.5, & \text{if } |j-i| = 1, i = 2, ..., (p-1), \\ 0.25, & \text{if } |j-i| = 2, i = 3, ..., (p-2), \\ 1, & \text{if } i = j, i = 1, ..., p, \\ 0, & \text{otherwise.} \end{cases}$$

Value

`x`	Simulated data in a *n*x*p* design matrix, without an intercept.
`y`	The response vector of dimension nx1. Quantitative for family='gaussian', binary (0-1) for family='binomial'. For family='cox', `y` should be an object of class `Surv`, as provided by the function `Surv()` in the package **survival**.
`A`	The true adjacency matrix of variables in the design matrix x.

Author(s)

Bochao Jia<`jbc409@gmail.com`> and Faming Liang

References

Liang, F., Xue, J. and Jia, B. (2018). Markov Neighborhood Regression for High-Dimensional Inference. Submitted to J. Amer. Statist. Assoc.

Examples

```
library(equSA)
p <- 200
coef_true <- rep(0,p)
coef_true[1:5] <- runif(5,3,5)
coef <- c(1,coef_true)
data <- SimMNR(n = 100, p = 200, coef = coef, family = "cox")
```

---

solcov *Calculate covariance matrix and precision matrix*

Description

Calculate the adjusted covriance matrix and precision matrix given the network structure from high dimesional dataset.

Usage

```
solcov(data, struct, tol=10^-5)
```

Arguments

data	A nxp data matrix.
struct	A preacquired adjacency matrix
tol	Tolerant value, default is 10^-5

Value

A list of two elements:

COV	Adjusted covriance matrix
PRE	Precision matrix

Author(s)

Bochao Jia<jbc409@gmail.com> & Runmin Shi

References

Friedman, J., Hastie, T., and Tibshirani, R. (2001). The elements of statistical learning (Vol. 1). Springer, Berlin: Springer series in statistics.

Examples

```
library(equSA)
data <- GauSim(100,200)
solcov(data$data,data$theta)
```

SR0 *One example dataset for ψ-learning alogorithm*

Description

SR0 is a simulated dataset for illustration our ψ-learning alogorithm.

Usage

```
data(SR0)
```

Format

SR0 dataset is a $100x200$ matrix. Each row represents a observation and each column represents a variable.

References

Liang, F., Song, Q. and Qiu, P. (2015). An Equivalent Measure of Partial Correlation Coefficients for High Dimensional Gaussian Graphical Models. J. Amer. Statist. Assoc., 110, 1248-1265.

TR0 *One example dataset for ψ-learning alogorithm*

Description

TR0 is a simulated dataset for illustration our ψ-learning alogorithm.

Usage

```
data(TR0)
```

Format

TR0 dataset is a $100x200$ matrix. Each row represents a observation and each column represents a variable.

References

Liang, F., Song, Q. and Qiu, P. (2015). An Equivalent Measure of Partial Correlation Coefficients for High Dimensional Gaussian Graphical Models. J. Amer. Statist. Assoc., 110, 1248-1265.

Index