Package 'ModPGMInference'

October 13, 2020

Type Package

Title Statistical Inference of Modified Poisson-Type Graphical Models

Version 1.0.0

Author Rong Zhang

Maintainer Rong Zhang <roz16@pitt.edu>

Description Provides a novel method for both edge-wise and global statistical inference with FDR control based on three modified Poisson-type graphical models: the truncated Poisson graphical model (TPGM), the sub-linear Poisson graphical model (SPGM) and the square-root Poisson graphical model (SqrtPGM). We focus on the high-dimensional settings where dimension p is allowed to be far larger than sample size n and implement the method using efficient proximal gradient descent. The method will be potentially useful in analysis of large count-valued or discrete omics data (e.g. RNA-seq). Other functions in the package including sample generation and data preprocessing for count-valued data. Windows users should install 'Rtools' before the installation of this package.

License GPL (>= 2)

Encoding UTF-8

URL https://github.com/zhangr100/ModPGMInference

LazyData true

Imports Rcpp (>= 1.0.5), RcppProgress, stats

LinkingTo Rcpp, RcppProgress

R topics documented:

ModPGMInference *Statistical Inference of Modified Poisson-Type Graphical Models*

Description

The main function for both edge-wise statistical inference and multiple testing with FDR control of three modified Poisson-type graphical models.

Usage

```
ModPGMInference <- function(x, model = "SqrtPGM", D = NULL, D_0 = NULL, D_1 = NULL,
tuning = "EBIC", gamma = NA_real_, kfold = NA_real_, nlambda = NA_real_,
step_size = NA_real_, intercept = TRUE, global = FALSE, alpha = NULL,
regularization = NULL, N = NA_real_, delta_upper = NA_real_, true_graph = NULL)
```

Arguments

x	An n by p data matrix (n is the sample size and p is the dimension, where p is allowed to be far larger than n).
model	Specification of modified Poisson-type graphical models: "TPGM" for truncated Poisson, "SPGM" for sub-linear Poisson and "SqrtPGM" for square-root Poisson. The default value is "SqrtPGM".
D	A p-length vector of truncation levels for each column of the input data matrix for "TPGM". All the values need to be positive. The default is a vector of maximum values for each column of the input data matrix.
D_0	A p-length vector of lower-bound truncation levels for each column of the input data matrix for "SPGM". The default is a vector with all 0s.
D_1	A p-length vector of upper-bound truncation levels for each column of the input data matrix for "SPGM". The default is a vector of maximum values for each column of the input data matrix.
tuning	Specification of methods for selection of tuning parameters when implementing the initial step of our approach: "EBIC" for the EBIC criterion and "CV" for the cross validation. The default value is "EBIC".
gamma	The parameter in the EBIC criterion. The default value is 0.5. ONLY applicable when tuning = "EBIC".
kfold	Specification of fold numbers in the cross validation. The default value is 5. ONLY applicable when tuning = "CV".
nlambda	Number of tuning parameters for each node-wise regression when implementing the initial step of our approach. The default value is 20.
step_size	A multiplicative parameter to decrease the step size during backtracking line search in the proximal gradient descent. Has to satisfy: 0 < step_size < 1. The default value is 0.5.
intercept	Should intercepts be estimated (intercept = TRUE) or set to 0s (intercept = FALSE) when implementing the initial step of our approach. The default is TRUE.
global	Should edge-wise (global = FALSE) or global statistical inference (global = TRUE) be performed based on the refined step of our approach. The default is FALSE.

alpha	A user-supplied sequence of pre-sepecified alpha levels for FDR control when global = TRUE. The default is alpha = 0.05,0.1 if no sequence is provided.
regularization	A user-supplied sequence of tuning parameters when only sole estimation is performed.
N	A pre-specified value related to the number (e.g. delta_upper*N) of δ values when selecting tuning parameters for global inference. The default value is 10.
delta_upper	A pre-specified value for the upper-bound level of δ (e.g. $0 \leq \delta \leq$ delta_upper) when selecting tuning parameters for global inference. The default value is 2.
true_graph	The true graph structure in a study if available. The default value is NULL. This argument is particularly for global inference. If a true graph is available, both FDR(s) and the corresponding power(s) will be provided in the outputs. Otherwise, only FDR(s) and the associated threshold(s) for all absolute values of test statistics will be provided.

Value

A list is returned including:

intercept	A sequence of estimated intercepts from the initial step of our approach.
theta_initial	A matrix of estimated θ_{ij}'s which depict conditional dependence of each $(i,j)^{th}$ pair of nodes after only the initial step of our approach.
theta_cor	A matrix of estimated θ_{ij}'s which depict conditional dependence of each $(i,j)^{th}$ pair of nodes after the refined step of our approach for bias correction.
CI_low_theta	A matrix of lower values of 95% confidence interval for each θ_{ij} based on the estimated values in theta_cor after bias correction.
CI_high_theta	A matrix of higher values of 95% confidence interval for each θ_{ij} based on the estimated values in theta_cor.
z_score	A matrix of z-scores for each θ_{ij} based on the estimated values in theta_cor.
p_thetaCor	A matrix of p-values for each θ_{ij} based on the estimated values in theta_cor.
est_sd	A matrix of estimated standard deviations for each θ_{ij} based on the estimated values in theta_cor.
threshold	The threshold sequence for absolute values of test statistics associated with the estimated FDR sequence.
FDR	The estimated FDR sequence for global inference of all pairs of θ_{ij}'s based on the pre-specified alpha level(s).
power	The estimated power sequence for global inference of all pairs of θ_{ij}'s associated with the estimated FDR sequence. ONLY applicable if true_graph is available.
global_decision	A list of p by p adjacency matrices of inferred graphs under the global inference corrsponding to the sequence of pre-sepecified alpha levels. A value of 1 in the matrix means that there is conditional dependence (or an edge) between the node pair, while a value of 0 means condtional independence (or no edge).

If regularization is available, only sole estimation is performed. Two lists named with intercept and theta_initial are returned, where each element records the estimated values corresponding to each user-defined tuning parameter with a descending order in the regularization sequence automatically.

Examples

```
## Chain graph
set <- c(-0.4,-0.3,-0.2,-0.1,0.1,0.2,0.3,0.4)
p <- 10

Omega.tmp <- matrix(0,p,p)
for(i in 1:(p-1)){
  j <- i+1
  Omega.tmp[i,j] <- sample(set,1)
}
for(i in 1:(p-1)){
  for(j in (i+1):p){
    Omega.tmp[j,i] <- Omega.tmp[i,j]
  }
}
Omega <- Omega.tmp

## Generate samples
n <- 100
X <- ModPGMSampler(psi = rep(0,p), true_graph = Omega, model = "TPGM", D = rep(3,p),
nSample = n, burn_in = 5000)

## Perform inference on random samples
result <- ModPGMInference(x = X, model = "TPGM", tuning = "EBIC", D = rep(3,p), nlambda = 100)
```

`ModPGMSampler` *Sample Generator for the Modified Poisson-Type Graphical Models*

Description

Implements sample generation for three modified Poisson-type graphical models based on the Gibbs sampler.

Usage

```
ModPGMSampler <- function(psi = NULL, true_graph, model = "SqrtPGM", D = NULL,
D_0 = NULL, D_1 = NULL, nSample, burn_in = NULL, thin = NULL)
```

Arguments

`psi`	A p-length vector of user-supplied values of intercepts. The default is a vector of all 0s if no sequence is provided.
`true_graph`	A known true graph structure.
`model`	Specification of modified Poisson-type graphical models: `"TPGM"` for truncated Poisson, `"SPGM"` for sub-linear Poisson and `"SqrtPGM"` for square-root Poisson. The default value is `"SqrtPGM"`.
`D`	A pre-specified p-length vector of truncation levels for each data column for `"TPGM"`. All the values need to be positive. The default is a vector of all 3s when there is no specification.
`D_0`	A pre-specified p-length vector of lower-bound truncation levels for each data column for `"SPGM"`. The default is a vector with all 3s when there is no specification.

D_1	A pre-specified p-length vector of upper-bound truncation levels for each data column for "SPGM". The default is a vector with all 6s when there is no specification.
nSample	The sample size n, or number of samples to be generated.
burn_in	The burn-in period of Gibbs sampler (or the number of samples to be discarded at the beginning). The default value is 1000.
thin	The thinning degree of Gibbs sampler. By default thin = 100 is used, which would result in keeping every 100th generated sample and discard all other samples.

Value

An n by p data matrix.

Examples

```
## Chain graph
set <- c(-0.4,-0.3,-0.2,-0.1,0.1,0.2,0.3,0.4)
p <- 10

Omega.tmp <- matrix(0,p,p)
for(i in 1:(p-1)){
  j <- i+1
  Omega.tmp[i,j] <- sample(set,1)
}
for(i in 1:(p-1)){
  for(j in (i+1):p){
    Omega.tmp[j,i] <- Omega.tmp[i,j]
  }
}
Omega <- Omega.tmp

## Generate samples
n <- 100
X <- ModPGMSampler(psi = rep(0,p), true_graph = Omega, model = "TPGM", D = rep(3,p),
nSample = n, burn_in = 5000)
```

ModPGM_true_sd *True Standard Deviation of Each Edge for Modified Poisson-Type Graphical Models*

Description

Returns true standard deviation of each θ_{ij} for three modified Poisson-type graphical models if a true graph structure is known.

Usage

```
ModPGM_true_sd <- function(x, psi = NULL, model = "SqrtPGM", true_graph, D = NULL,
D_0 = NULL, D_1 = NULL)
```

Arguments

x	An n by p data matrix (n is the sample size and p is the dimension).
psi	A p-length vector of user-supplied values of intercepts. The default is a vector of all 0s if no sequence is provided.
model	Specification of modified Poisson-type graphical models: "TPGM" for truncated Poisson, "SPGM" for sub-linear Poisson and "SqrtPGM" for square-root Poisson. The default value is "SqrtPGM".
true_graph	A known true graph structure.
D	A pre-specified p-length vector of truncation levels for each data column for "TPGM". All the values need to be positive. The default is a vector of maximum values for each column of the input data matrix.
D_0	A pre-specified p-length vector of lower-bound truncation levels for each data column for "SPGM". The default is a vector with all 0s.
D_1	A pre-specified p-length vector of upper-bound truncation levels for each data column for "SPGM". The default is a vector of maximum values for each column of the input data matrix.

Value

A p by p matrix which encodes true standard deviations from all node pairs.

Examples

```
## Chain graph
set <- c(-0.4,-0.3,-0.2,-0.1,0.1,0.2,0.3,0.4)
p <- 10

Omega.tmp <- matrix(0,p,p)
for(i in 1:(p-1)){
  j <- i+1
  Omega.tmp[i,j] <- sample(set,1)
}
for(i in 1:(p-1)){
  for(j in (i+1):p){
    Omega.tmp[j,i] <- Omega.tmp[i,j]
  }
}
Omega <- Omega.tmp

## Generate samples
n <- 100
X <- ModPGMSampler(psi = rep(0,p), true_graph = Omega, model = "TPGM", D = rep(3,p),
nSample = n, burn_in = 5000)

## Obtain true standard deviation of each edge
true_sd <- ModPGM_true_sd(x = X, psi = rep(0,p), model = "TPGM", true_graph = Omega,
D = rep(3,p))
```

Preprocess *Preprocessing the Count-Valued RNA-Seq Data*

Description

Performs preprocessing on raw count-valued data (e.g. RNA-seq).

Usage

```
Preprocess <- function(X, quanNorm = 0.75, nLowCount = 20, percentLowCount = 0.95,
NumGenes = 500, log_power_trans_only = FALSE)
```

Arguments

X	An n by p raw count matrix.
quanNorm	A parameter to control the quantile normalization for the data. The default value is 0.75.
nLowCount	The minimum count value across samples to filter out a gene. The default value is 20.
percentLowCount	The percent of samples for a gene to be less than nLowCount. The default value is 0.95.
NumGenes	Number of genes to keep in the data after preprocessing. The default value is 500.
log_power_trans_only	Only log or power transform is performed and retains NumGenes genes with largest inter-sample variances if log_power_trans_only = TRUE. The default is FALSE.

Details

The preprocessing steps for the raw count-valued data include: 1. Quantile normalization for samples; 2. Filter out genes with low counts across all samples; 3. Retain genes with large inter-sample variances; 4. Use log or power transform which is selected based on Kolmogorov-Smirnov goodness of fit test to make the data closer to the Poisson distribution.

Value

An n by NumGenes or p or number of retained genes data matrix after preprocessing.

Examples

```
## Chain graph
set <- c(-0.4,-0.3,-0.2,-0.1,0.1,0.2,0.3,0.4)
p <- 20

Omega.tmp <- matrix(0,p,p)
for(i in 1:(p-1)){
  j <- i+1
  Omega.tmp[i,j] <- sample(set,1)
}
```

```
for(i in 1:(p-1)){
  for(j in (i+1):p){
    Omega.tmp[j,i] <- Omega.tmp[i,j]
  }
}
Omega <- Omega.tmp

## Simulate a count-valued data set
n <- 100
X <- ModPGMSampler(psi = rep(0,p), true_graph = Omega, model = "TPGM", D = rep(3,p),
nSample = n, burn_in = 5000)

uniform <- matrix(0,n,p)
for(k in 1:n){
  uniform[k,] <- runif(p,0,1)
}

X_new <- X + uniform

count_value <- exp(log(X_new)/0.2517)

count_value <- floor(count_value)

## Pre-processing
X_process <- Preprocess(X = count_value)
```

Index