Package ‘BICORN’

January 20, 2025

Title Integrative Inference of De Novo Cis-Regulatory Modules

Version 0.1.0

Description Prior transcription factor binding knowledge and target gene expression data are integrated in a Bayesian framework for functional cis-regulatory module inference. Using Gibbs sampling, we iteratively estimate transcription factor associations for each gene, regulation strength for each binding event and the hidden activity for each transcription factor.

Depends R (>= 3.4)

License GPL-2

Encoding UTF-8

LazyData true

RoxygenNote 6.0.1

Suggests knitr, rmarkdown

Imports stats

VignetteBuilder knitr

NeedsCompilation no

Author Xi Chen [aut, cre],
Jianhua Xuan [aut]

Maintainer Xi Chen <`xichen86@vt.edu`>

Repository CRAN

Date/Publication 2018-06-06 10:30:06 UTC

Contents

`A` *TF-gene regulation strength matrix*

Description

A matrix of TF-gene regulation strength with genes as rows and TFs as columns.

Usage

```
A
```

Format

numeric matrix

`alpha` *Inverse-gamma distribution hyper-parameter alpha*

Description

Hyper-parameter alpha of inverse-gamma distribution.

Usage

```
alpha
```

Format

scalar

`A_old` *TF-gene regulation strength matrix sampled from the previous round*

Description

A matrix of TF-gene regulation strength with genes as rows and TFs as columns, sampled from the previous round. During the Gibbs sampling process, this matrix is used as prior for a new round of regulation strength sampling.

Usage

```
A_old
```

Format

numeric matrix

`A_sampling` *Regulation Strength Sampling Function*

Description

Function 'A_sampling' estimates a regulation strength for each sampled binding event in C, according to a posterior Gaussian distribution.

Usage

```
A_sampling(Y, C, A_old, X, base_line, C_prior, sigma_noise, sigma_A,
  sigma_baseline, sigma_X)
```

Arguments

`Y`	gene expression data matrix
`C`	sampled TF-gene binding network
`A_old`	regulatory strength sampled from the previous round, used as a prior in current function
`X`	sampled transcription factor activity matrix
`base_line`	sampled gene expression baseline activity
`C_prior`	prior TF-gene binding network
`sigma_noise`	variance of gene expression fitting residuals
`sigma_A`	variance of regulatory strength
`sigma_baseline`	variance of gene expression baseline activity
`sigma_X`	variance of transcription factor activity

`baseline_sampling` *Gene Baseline Expression Sampling Function*

Description

Function 'baseline_sampling' estimates a baseline expression for each gene, according to a posterior Gaussian distribution.

Usage

```
baseline_sampling(Y, C, A, X, base_line_old, C_prior, sigma_noise, sigma_A,
  sigma_baseline, sigma_X)
```

Arguments

`Y`	gene expression data matrix
`C`	sampled TF-gene binding network
`A`	sampled regulatory strength matrix
`X`	sampled transcription factor activity matrix
`base_line_old`	prior gene expression baseline activity
`C_prior`	prior TF-gene binding network
`sigma_noise`	variance of gene expression fitting residuals
`sigma_A`	variance of regulatory strength
`sigma_baseline`	variance of gene expression baseline activity
`sigma_X`	variance of transcription factor activity

`base_line` *Gene baseline expression*

Description

A vector of baseline expression for all genes.

Usage

```
base_line
```

Format

numeric vector

`base_line_old` *Gene baseline expression sampled from the previous round.*

Description

A vector of baseline expression for all genes, sampled from the previous round. During the Gibbs Samplig process, this is used as a prior for a new round of gene baseline expression sampling.

Usage

```
base_line_old
```

Format

numeric vector

`beta` *Inverse-gamma distribution hyper-parameter beta*

Description

Hyper-parameter beta of inverse-gamma distribution.

Usage

```
beta
```

Format

scalar

`BICORN` *BICORN Algorithm Function*

Description

Function 'BICORN' infers a posterior module-gene regulatory network by iteratively sampling regulatory strength, transcription factor activity and several key model parameters.

Usage

```
BICORN(BICORN_input = NULL, L = 100, output_threshold = 10)
```

Arguments

`BICORN_input`	this list structure contains TF symbols, gene symbols and candidate modules
`L`	total rounds of Gibbs Sampling.
`output_threshold`	number of rounds after which we start to record results.

Examples

```
# load in the sample data input
data("sample.input")

# Data initialization (Integerate prior binding network and gene expression data)
BICORN_input<-data_integration(Binding_matrix = Binding_matrix, Binding_TFs = Binding_TFs,
Binding_genes = Binding_genes, Exp_data = Exp_data, Exp_genes = Exp_genes,
Minimum_gene_per_module_regulate = 2)

# Infer cis-regulatory modules (TF combinations) and their target genes
BICORN_output<-BICORN(BICORN_input, L = 2, output_threshold = 1)
```

`Binding_genes` *Genes in the prior binding network*

Description

A list of offical gene symbols in the binary binding network.

Usage

```
Binding_genes
```

Format

character vector

`Binding_matrix` *Prior TF-gene binding network*

Description

A prior binary TF-gene regulatory network with each unit either 1 (binding) or 0 (non-binding).

Usage

```
Binding_matrix
```

Format

numeric matrix

`Binding_TFs` *TFs in the prior binding network*

Description

A list of transcription factors in the prior binding network.

Usage

```
Binding_TFs
```

Format

character vector

`C` *TF-gene binding network*

Description

A matrix of TF-gene regulatory network with each unit either 1 (binding) or 0 (non-binding).

Usage

```
C
```

Format

numeric matrix

`C_old` *TF-gene binding network sampled from the previous round*

Description

A matrix of TF-gene binding network sampled from the previous round, with each unit either 1 (binding) or 0 (non-binding). During the Gibbs sampling process, this is used as a prior for a new round of binding network sampling.

Usage

```
C_old
```

Format

numeric matrix

`C_prior` *Prior TF-gene binding network*

Description

A matrix of prior TF-gene binding events, with each unit either 1 (binding) or 0 (non-binding). Such a prior network can be obtained from TF-gene binding database, motif searching, ChIP-seq peaks or ATAC-seq peaks.

Usage

```
C_prior
```

Format

numeric matrix

`C_sampling_cluster` *cis-Regulatory Module Sampling Function*

Description

Function 'C_sampling_cluster' samples a candidate cis-regulatory module for each gene, according to a discrete posterior probability distribution.

Usage

```
C_sampling_cluster(Y, C_old, A_old, X_old, base_line_old, C_prior, sigma_noise,
  sigma_A, sigma_baseline, sigma_X, BICORN_input)
```

Arguments

`Y`	gene expression data matrix
`C_old`	TF-gene binding network sampled from the previous round
`A_old`	regulatory strength matrix sampled from the previous round
`X_old`	transcription factor activity matrix sampled from the previous round
`base_line_old`	gene expression baseline activity sampled from the previous round
`C_prior`	prior TF-gene binding network
`sigma_noise`	variance of gene expression fitting residuals
`sigma_A`	variance of regulatory strength
`sigma_baseline`	variance of gene expression baseline activity
`sigma_X`	variance of transcription factor activity
`BICORN_input`	this list structure contains TF symbols, Gene symbols and candidate modules

---

`data_integration` *Data Initialization for BICORN*

Description

Function 'data_integration' integrates the prior TF-gene binding network and gene expression data together. It will remove any genes missing either TF bindings or gene expression and identify a list of candidate cis-regulatory modules.

Usage

```
data_integration(Binding_matrix = NULL, Binding_TFs = NULL,
  Binding_genes = NULL, Exp_data, Exp_genes = NULL,
  Minimum_gene_per_module_regulate = 2)
```

Arguments

`Binding_matrix`	loaded prior binding network
`Binding_TFs`	loaded transcription factors
`Binding_genes`	loaded genes in the prior binding network
`Exp_data`	loaded properly normalized gene expression data
`Exp_genes`	loaded genes in the gene expression data
`Minimum_gene_per_module_regulate`	the minimum number of genes regulated by each module, used for candidate module filtering.

Examples

```
# load in the sample data input
data("sample.input")

# Data initialization (Integerate prior binding network and gene expression data)
BICORN_input<-data_integration(Binding_matrix = Binding_matrix, Binding_TFs = Binding_TFs,
Binding_genes = Binding_genes, Exp_data = Exp_data, Exp_genes = Exp_genes,
Minimum_gene_per_module_regulate = 2)
```

`Exp_data` *Gene expression data*

Description

A matrix of normalized gene expression data with genes as rows and samples as columns. The gene expression data can be either time-course data measured under multiple time points or steady state data generated from at least two different conditions.

Usage

```
Exp_data
```

Format

numeric matrix

`Exp_genes` *Genes in the expression data*

Description

A list of official gene symbols in the gene expression data set.

Usage

```
Exp_genes
```

Format

character vector

`sigmanoise_sampling` *Fitting Residule Variance Sampling Function*

Description

Function 'sigmanoise_sampling' estimates the variance of overal gene expression fitting residuals, according to an inverse-gamma distribution.

Usage

```
sigmanoise_sampling(Y, C, A, X, base_line, C_prior, sigma_A, sigma_baseline,
  sigma_X, alpha, beta)
```

Arguments

`Y`	gene expression data matrix
`C`	sampled TF-gene binding network
`A`	sampled regulatory strength matrix
`X`	sampled transcription factor activity matrix
`base_line`	sampled gene expression baseline activity
`C_prior`	prior TF-gene binding network
`sigma_A`	variance of regulatory strength
`sigma_baseline`	variance of gene expression baseline activity
`sigma_X`	variance of transcription factor activity
`alpha`	hyper-parameter for inverse-gamma distribution
`beta`	hyper-parameter for inverse-gamma distribution

`sigma_A` *Regulation strength variance*

Description

Variance of regulation strength matrix A.

Usage

```
sigma_A
```

Format

scalar

`sigma_baseline` *Variance of baseline gene expression.*

Description

Variance of baseline gene expression.

Usage

```
sigma_baseline
```

Format

scalar

`sigma_noise` *Variance of gene expression fitting residuals.*

Description

Variance of gene expression fitting residuals.

Usage

```
sigma_noise
```

Format

scalar

`sigma_X` *Transcription factor activity variance*

Description

Variance of transcription factor activity matrix X.

Usage

```
sigma_X
```

Format

scalar

`X` *Transcription factr activity matrix*

Description

A matrix of hidden transcription factr activity estimated from gene expression data, with transcription factrs as rows and samples as columns.

Usage

```
X
```

Format

numeric matrix

`X_old` *Transcription factr activity matrix sampled from the previous round*

Description

A matrix of hidden transcription factr activity estimated from gene expression data, with transcription factrs as rows and samples as columns, sampled from the previous round. During the Gibbs sampling process, this is used as a prior for a new round of transcription factor activity sampling.

Usage

```
X_old
```

Format

numeric matrix

`X_sampling` *Transcription Factor Activity Sampling Function*

Description

Function ’X_sampling’ estimates the hidden activities of each transcription factor, according to a posterior Gaussian random process.

Usage

```
X_sampling(Y, C, A, X_old, base_line, C_prior, sigma_noise, sigma_A,
  sigma_baseline, sigma_X)
```

Arguments

`Y`	gene expression data matrix
`C`	sampled TF-gene binding network
`A`	sampled regulatory strength matrix
`X_old`	sampled transcription factor activity matrix from the previous round
`base_line`	sampled gene expression baseline activity
`C_prior`	prior TF-gene binding network
`sigma_noise`	variance of gene expression fitting residuals
`sigma_A`	variance of regulatory strength
`sigma_baseline`	variance of gene expression baseline activity
`sigma_X`	variance of transcription factor activity

`Y` *Gene expression data used for module inference*

Description

A matrix of normalized gene expression for common genes of prior binding input and gene expression input, with genes as rows and samples as columns. Y is the matrix used for cis-regulatory mudole inference.

Usage

```
Y
```

Format

numeric matrix

Index