Package 'birta'

April 15, 2019

Version 1.26.0

Date 2013-03-07

Title Bayesian Inference of Regulation of Transcriptional Activity

Author Benedikt Zacher, Khalid Abnaof, Stephan Gade, Erfan Younesi, Achim Tresch, Holger Froehlich

Maintainer Benedikt Zacher <zacher@lmb.uni-muenchen.de>, Holger Froehlich <frohlich@bit.uni-bonn.de>

Depends limma, MASS, R(>= 2.10), Biobase, methods

Description Expression levels of mRNA molecules are regulated by different processes, comprising inhibition or activation by transcription factors and post-transcriptional degradation by microRNAs. birta (Bayesian Inference of Regulation of Transcriptional Activity) uses the regulatory networks of TFs and miRNAs together with mRNA and miRNA expression data to predict switches in regulatory activity between two conditions. A Bayesian network is used to model the regulatory structure and Markov-Chain-Monte-Carlo is applied to sample the activity states.

License GPL (>= 2)

LazyLoad yes

biocViews Microarray, Sequencing, GeneExpression, Transcription, GraphAndNetwork

git_url https://git.bioconductor.org/packages/birta

git_branch RELEASE_3_8

git_last_commit 43cffee

git_last_commit_date 2018-10-30

Date/Publication 2019-04-15

R topics documented:

`birta-package` *Joint Bayesian Inference of miRNA and Transcription Factor Activities*

Description

Expression levels of mRNA molecules are regulated by different processes, comprising inhibition or activation by transcription factors and post-transcriptional degradation by microRNAs. birta uses the regulatory networks of TFs and miRNAs together with mRNA and miRNA expression data to predict switches in regulatory activity between two conditions. A Bayesian network is used to model the regulatory structure and Markov-Chain-Monte-Carlo is applied to sample the activity states.

Details

Package:	birta
Type:	Package
Version:	0.99.1
Date:	2012-03-6
License:	GNU GENERAL PUBLIC LICENSE (GPL) >= 2
LazyLoad:	yes

Author(s)

Benedikt Zacher, Khalid Abnaof, Stephan Gade, Erfan Younesi, Achim Tresch, Holger Frohlich
Maintainer: Benedikt Zacher <zacher@lmb.uni-muenchen.de>

References

B. Zacher, K. Abnaof, S. Gade, E. Younesi, A. Tresch, and H. Frohlich. Joint Bayesian Inference of Condition Specific miRNA and Transcription Factor Activities from Combined Gene and microRNA Expression Data. *submitted*, 2012.

birta *Main interface for Bayesian Inference of Regulation of Transcriptional Activity.*

Description

The function estimates parameterization of the model and then executes MCMC sampling to infer activity states.

Usage

```
birta(dat.mRNA, dat.miRNA, TFexpr, limmamRNA=NULL, limmamiRNA=NULL, limmaTF=NULL, nrep=NULL, fdr.
```

Arguments

dat.mRNA	mRNA expression data (ExpressionSet or matrix). IMPORTANT: Replicates must be ordered according to nrep!
dat.miRNA	miRNA expression data (ExpressionSet or matrix).
TFexpr	TF expression data (ExpressionSet or matrix).
limmamRNA	Output of limma analysis for mRNA data (list: pvalue.tab, lm.fit).
limmamiRNA	Output of limma analysis for miRNA data (list: pvalue.tab, lm.fit).
limmaTF	Output of limma analysis for TF expression data (list: pvalue.tab, lm.fit).
nrep	Vector containing four integers. Entry 1 and 2 are the number of miRNA measurement replicates for condition 1 and 2. Entry 3 and 4 are the number of mRNA measurement replicates for condition 1 and 2.
fdr.mRNA	False discovery rate (FDR) cutoff for significance of the logFC for mRNA data.
fdr.miRNA	False discovery rate (FDR) cutoff for significance of the logFC for miRNA data.
lfc.mRNA	Additional logFC cutoff for significance in mRNA data.
lfc.miRNA	Additional logFC cutoff for significance in miRNA data.
genesets	Combined TF / miRNA network. IMPORTANT: Names of TF entries must start with V\$.
lambda	Regularization parameter for edge weights.
sample.weights	Should edge weights be adapted during sampling?
one.regulator.weight	Sould weights of all edges for a regulator to be the same?
theta_TF	Expected fraction of active TFs.
theta_miRNA	Expected fraction of active miRNAs.
model	Type of model. One out of c("all-plug-in", "weight-plug-in", "no-plug-in").
niter	Number of MCMC iterations (AFTER burnin).
nburnin	Number of MCMC iterations UNTIL burnin is assumed to be finished.
potential_swaps	Pre-computed potential swaps (OPTIONAL, see get_potential_swaps).
run.pretest	Initialize miRNA and TF states via the result of a hypergeometric test in order to improve convergence (should be taken with care; advise: only use it in case of observed convergence problems!).

`condition.specific.inference`
: Should inference on TF / miRNA activities be made only RELATIVE to a reference condition or independently in both conditions?

`thin`
: Thinning of Markov chain: only use every thin's sample for posterior computation.

`only_switches`
: Should only switches be performed?

`weightSampleMean`
: Mean for normal distribution used for sampling the omegas.

`weightSampleVariance`
: Variance for normal distribution used for sampling the omegas.

Value

The function returns a list containing the following entries:

`miRNAstates1`
: Probability of each miRNA to be active in condition 1 (only for condition specific sampling).

`miRNAstates2`
: Probability of each miRNA to be active in condition 2 (only for condition specific sampling).

`miRNAActivitySwitch`
: Probability of each miRNA switching its activities (non-condition specific sampling).

`TFstates1`
: Probability of each TF to be active in condition 1 (only for condition specific sampling).

`TFstates2`
: Probability of each TF to be active in condition 2 (only for condition specific sampling).

`miRNAactivitySwitch`
: Probability of each TF switching its activities (non-condition specific sampling).

`log_lik_trace`
: Log-likelihood trace of MCMC sampling.

`TFomega`
: Weights of TF-target graph. (effect of a TF on its targets)

`miRNAomega`
: Weights of miRNA-target graph. (effect of a miRNA on its targets)

`genesetsTF`
: TF-target network. This might be different from the network submitted to the function, due to incompatibilities of network and experimental measurements. Check your warnings and command line output!

`genesetsmiRNA`
: miRNA-target network. This might be different from the network submitted to the function, due to incompatibilities of network and experimental measurements. Check your warnings and command line output!

`mRNAexpr`
: mRNA expression data. This might be different from the matrix submitted to the function, due to incompatibilities of network and experimental measurements. Check your warnings and command line output!

`miRNAexpr`
: miRNA expression data. This might be different from the matrix submitted to the function, due to incompatibilities of network and experimental measurements. Check your warnings and command line output!

`TFexpr`
: TF expression data (only if they are specifically included). This might be different from the matrix submitted to the function, due to incompatibilities of network and experimental measurements. Check your warnings and command line output!

Author(s)

Holger Frohlich, Benedikt Zacher

Examples

```
data(humanSim)
design = model.matrix(~0+factor(c(rep("control", 5), rep("treated", 5))))
colnames(design) = c("control", "treated")
contrasts = "treated - control"
limmamRNA = limmaAnalysis(sim$dat.mRNA, design, contrasts)
limmamiRNA = limmaAnalysis(sim$dat.miRNA, design, contrasts)
sim_result = birta(sim$dat.mRNA, sim$dat.miRNA, limmamRNA=limmamRNA,
 limmamiRNA=limmamiRNA, nrep=c(5,5,5,5), genesets=genesets,
 model="all-plug-in", niter=50000, nburnin=10000,
 sample.weights=FALSE, potential_swaps=potential_swaps)
```

`birta-methods` *Methods for Function* `birta` *in Package* **birta**

Description

Generic methods for function `birta` in package **birta**. The expression data can be stored in a matrix or an ExpressionSet.

Methods

The following signatures make sure, that the arguments, storing the expression data are in the correct format.

```
signature(dat.mRNA = "ExpressionSet", dat.miRNA = "ExpressionSet", TFexpr = "ExpressionSet")
signature(dat.mRNA = "ExpressionSet", dat.miRNA = "missing", TFexpr = "ExpressionSet")
signature(dat.mRNA = "ExpressionSet", dat.miRNA = "missing", TFexpr = "missing")
signature(dat.mRNA = "matrix", dat.miRNA = "matrix", TFexpr = "matrix")
signature(dat.mRNA = "matrix", dat.miRNA = "matrix", TFexpr = "missing")
signature(dat.mRNA = "matrix", dat.miRNA = "missing", TFexpr = "matrix")
signature(dat.mRNA = "matrix", dat.miRNA = "missing", TFexpr = "missing")
```

EColiNetwork *Example TF-target graph from Regulon DB.*

Description

This list contains the TF-target graph used in the vignette.

Usage

```
EColiNetwork
```

Format

A list containing the target gene sets of 160 TFs.

Source

This TF-target graph was taken from (R. Castelo and A. Roverato, 2009). It is a pre-filtered version of RegulonDB 6.1.

References

R. Castelo and A. Roverato. Reverse engineering molecular regulatory networks from microarray data with qp-graphs. *J Comput Biol*, 16(2):213227, Feb 2009.

EColiOxygen *Example data set from E. Coli to sample TF activities.*

Description

This data set gives expression values for three experiments of the E. Coli K12 strain under aerobic and three experiments under aerobic growth. It is used in the vignette to illustrate application of birta to TFs only.

Usage

```
EColiOxygen
```

Format

ExpressionSet

Source

The original data comes from (Covert et al., 2004) The normalized data set used here is taken from the qpgraph package by R. Castelo and A. Roverato.

References

M. W. Covert, E. M. Knight, J. L. Reed, M. J. Herrgard, and B. O. Palsson. Integrating high-throughput and computational data elucidates bacterial networks. *Nature*, 429(6987):9296,May 2004.

genesets *TF-target and miRNA-target networks for simulated example.*

Description

For miRNAs we looked at target predictions in human via miRanda (Betel et al., 2008), miRBase (Griffths-Jones et al., 2008) and miRDB (Wang and Naqa, 2008). Experimentally confirmed targets were retrieved from Tarbase (Papadopoulos et al., 2009). As trustworthy miRNA targets we considered those, which were either experimentally confirmed (i.e. listed in Tarbase) or predicted by at least two of the other three methods. In conclusion we arrived at a network with 583 miRNAs regulating between 1 and 1628 genes (median: 302).\ A TF-target gene network was compiled by computing TF binding affinities to promoter sequences of all human genes according to the TRAP model (Roideret al., 2007) via the author's R implementation. Upstream sequences of genes were retrieved here from the ENSEMBL database via biomaRt (Haider et al., 2009). We assumed that promoter sequences were located in the range 0 - 2Kbp upstream to the transcription start site of a gene. As trustworthy TF targets we considered those, for which a Holm corrected affinity p-value smaller than 0.001 was reported. In conclusion we arrived at a network with 272 TFs regulating between 1 and 16517 genes (median: 20).

Usage

```
genesets
```

Format

A list containing a TF- and miRNA-target graph.

Source

The networks were generated using the TRAP model on TRANSFAC matrices and miRNA-target annotations form different databases.

References

D. Betel, M. Wilson, A. Gabow, D. S. Marks, and C. Sander. The microrna.org resource: targets and expression. *Nucleic Acids Res*, 36(Database issue):D149-D153, Jan 2008.

S. Griffiths-Jones, H. K. Saini, S. van Dongen, and A. J. Enright. miRBase: tools for microRNA genomics. *Nucleic Acids Res*, 36(Database issue):D154-D158, Jan. 2008.

X. Wang and I. M. E. Naqa. Prediction of both conserved and nonconserved microrna targets in animals. *Bioinformatics*, 24(3):325-332, Feb 2008.

G. L. Papadopoulos, M. Reczko, V. A. Simossis, P. Sethupathy, and A. G. Hatzigeorgiou. The database of experimentally supported targets: a functional update of tarbase. *Nucleic Acids Res*, 37(Database issue):D155-D158, Jan 2009.

get_potential_swaps *Calculate swap partner for TF-/miRNA-target graph.*

Description

Calculates for TF-/miRNA-target graph all potential swap partner.

Usage

```
get_potential_swaps(genesetsTF=NULL, genesetsmiRNA=NULL, perc.overlap.cutoff = 0.8, integer.id=TR
```

Arguments

genesetsTF	Each entry corresponds to a TF and contains its target genes.
genesetsmiRNA	Each entry corresponds to a miRNA and contains its target genes.
perc.overlap.cutoff	Percentage cutoff of minimal overlap between two miRNAs or TFs to be possible swap partner.
integer.id	If TRUE, the swap partner are not outputted as characters, but as integer indices.
verbose	print process or not.

Value

The function returns a list, with the following entries:

T_potential_swaps	Contains for each TF all potential swap partner.
S_potential_swaps	Contains for each miRNA all potential swap partner.

Author(s)

Benedikt Zacher (zacher@lmb.uni-muenchen.de)

Examples

```
genesetsTF = list("V$1"=c("A", "B", "C", "D", "E"), "V$2"=c("A", "B", "C", "D"), "V$3"=c("A", "B", "C"))
genesetsmiRNA =list("miR-1"=c("C", "E", "D", "G", "H"), "miR-2"=c("C", "E", "D", "G"), "miR-3"=c("C", "E"))
get_potential_swaps(genesetsTF, genesetsmiRNA, integer.id=FALSE, perc.overlap.cutoff=0.7)
```

`limmaAnalysis` *Perform a limma analysis on expression data.*

Description

Needed for the parameterization of the model as well as the Fisher (pre-)test.

Usage

```
limmaAnalysis(dat, design, contrasts)
```

Arguments

`dat`	A matrix or ExpressionSet containing the expression values.
`design`	A design matrix.
`contrasts`	Contrast for the linear model.

Value

Returns a list containing the following entries:

`pvalue.tab`	Containing the result of the topTable function from limma.
`lm.fit`	Linear fit to the model.
`design`	The design used.
`contrast`	The contrasts used.

Author(s)

Benedikt Zacher (zacher@lmb.uni-muenchen.de), Holger Frohlich

References

G. K. Smyth. Limma : Linear Models for Microarray Data. *Bioinformatics*, (2005):397-420.

See Also

`lmFit`, `topTable`

Examples

```
data(humanSim)
design = model.matrix(~0+factor(c(rep("control", 5), rep("treated", 5))))
colnames(design) = c("control", "treated")
contrasts = "treated - control"
limmamRNA = limmaAnalysis(sim$dat.mRNA, design, contrasts)
```

limmaAnalysis-methods *Methods for Function* limmaAnalysis *in Package* **birta**

Description

Generic methods for function limmaAnalysis in package **birta**. The expression data can be stored in a matrix or an ExpressionSet.

Methods

signature(dat = "ExpressionSet", design = "matrix", contrasts = "character") Generic function for objects of class ExpressionSet.

signature(dat = "matrix", design = "matrix", contrasts = "character") Generic function for objects of class matrix.

plotConvergence *Plotting the likelihood along MCMC sampling.*

Description

Plots the log likelihood along MCMC sampling.

Usage

```
plotConvergence(res, nburnin=NULL, title="")
```

Arguments

res	The result from birta.run (a list).
nburnin	Number of iterations used for the burn in.
title	Optional title of the plot.

Author(s)

Benedikt Zacher <zacher@lmb.uni-muenchen.de>

See Also

birta

Examples

```
data(humanSim)
data(humanSim)
design = model.matrix(~0+factor(c(rep("control", 5), rep("treated", 5))))
colnames(design) = c("control", "treated")
contrasts = "treated - control"
limmamRNA = limmaAnalysis(sim$dat.mRNA, design, contrasts)
limmamiRNA = limmaAnalysis(sim$dat.miRNA, design, contrasts)
sim_result = birta(sim$dat.mRNA, sim$dat.miRNA, limmamRNA=limmamRNA,
 limmamiRNA=limmamiRNA, nrep=c(5,5,5,5), genesets=genesets,
 model="all-plug-in", niter=50000, nburnin=10000,
 sample.weights=FALSE, potential_swaps=potential_swaps)
plotConvergence(sim_result, nburnin=10000, title="simulation")
```

---

potential_swaps *Potential swap moves for TF-target and miRNA-target networks (see genestes data set).*

Description

Potential swap moves for TF-target and miRNA-target networks (see genestes data set) used for the simulation in the vignette.

Usage

```
potential_swaps
```

Format

A list containing potential swap moves for the TF- and miRNA-target graph used in the simulation.

---

sim *Simulated expression data for mRNAs and miRNAs.*

Description

This data set contains simulated expression values of miRNAs and mRNAs, together with the associated TF- and miRNA-target networks.

Usage

```
sim
```

Format

A list containing the entries dat.mRNA (mRNA expression), dat.miRNA (miRNA expression), miRNAstates (activity states of miRNAs), TFstates (activity states of TFs, TRANSFAC), genesetsmiRNA (miRNA-target graph, targetScan), genesetsTFs (TF-taget graph).

Source

Simulation (see reference for details).

References

B. Zacher, K. Abnaof, S. Gade, E. Younesi, A. Tresch, and H. Frohlich. Joint Bayesian Inference of Condition Specific miRNA and Transcription Factor Activities from Combined Gene and microRNA Expression Data. *submitted*, 2012.

`TFexpr` *Transcription factor expression values for the aerobic-anaerobic growth experiment.*

Description

This data set gives expression values for the 160 TF of the TF-target graph EColiNetwork used in the vignette.

Usage

```
TFexpr
```

Format

ExpressionSet. Rownames in the assayData correspond to entries in TF-target graph.

Source

See EColiOxygen and EColiNetwork (see reference for details).

References

M. W. Covert, E. M. Knight, J. L. Reed, M. J. Herrgard, and B. O. Palsson. Integrating high-throughput and computational data elucidates bacterial networks. *Nature*, 429(6987):92-96, May 2004.

Index