Package 'TFARM'

December 5, 2025

Type Package

Title Transcription Factors Association Rules Miner

Version 1.33.0

Date 2021-11-21

Depends R (>= 3.4)

Author Liuba Nausicaa Martino, Alice Parodi, Gaia Ceddia, Piercesare Secchi, Stefano Campaner, Marco Masseroli

Maintainer Liuba Nausicaa Martino <liuban.martino@gmail.com>

Description It searches for relevant associations of transcription factors with a transcription factor target, in specific genomic regions.
It also allows to evaluate the Importance Index distribution of transcription factors (and combinations of transcription factors) in association rules.

License Artistic-2.0

Imports arules, fields, GenomicRanges, graphics, stringr, methods, stats, gplots

VignetteBuilder knitr

Suggests BiocStyle, knitr, plyr

biocViews BiologicalQuestion, Infrastructure, StatisticalMethod, Transcription

LazyData TRUE

RoxygenNote 6.1.1

NeedsCompilation no

Encoding UTF-8

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

git_branch devel

git_last_commit cbabd4a

git_last_commit_date 2025-10-29

Repository Bioconductor 3.23

Date/Publication 2025-12-04

Contents

DELTA . . . 2
distribViz . . . 3
heatI . . . 3
IComp . . . 4
IMP . . . 5
imp_FOSL2 . . . 6
IMP_Z . . . 6
IPCA . . . 7
items . . . 8
itemset . . . 8
I_c_2 . . . 9
MCF7_chr1 . . . 9
presAbs . . . 10
p_TFs . . . 11
rulesGen . . . 11
rulesTF . . . 12
rulesTF0 . . . 13
r_FOSL2 . . . 14
r_noFOSL2 . . . 15
r_TEAD4 . . . 15
search_rule . . . 16
TF_Imp . . . 16

Index **18**

`DELTA` *Contains the delta variations of support, confidence and lift.*

Description

DELTA is a list of 12 elements and each element has two columns representing support (`diff_supp_Z`) and confidence (`diff_conf_Z`) respectively. It is included in the `data_man` collection.

Usage

```
data("data_man")
```

Format

An object of class `"list"`

Examples

```
# DELTA is found in the data_man collection of datasets:
data("data_man")
head(DELTA[[1]])
```

distribViz — *Boxplots visualization of the Importance Index distribution of a set of transcription factors.*

Description

For a set of candidate co-regulator transcription factors, the Importance Index distribution of each transcription factor is plotted in boxplots. The shape of every boxplot depends on: the dimension of the distribution, which is equal to the number of rules in which each transcription factor appears (the higher is such number, the larger is a boxplot), the variability of the distribution (the higher is the variability of the Importance Index distribution, the longer is a boxplot). Moreover, the higher is the median of the Importance Index distribution for a candidate co-regulator transcription factor, the higher the boxplot is aligned with respect to the y-axis.

Usage

```
distribViz(I, TFs)
```

Arguments

I	a list of Importance Index distributions (IIDs)
TFs	string vector with the names of the transcription factors

Value

A boxplot representing the transcription factor IIDs

Examples

```
# Load IMP_Z and p from the data_man collection of datasets:
data('data_man')

# Plot the Importance Index distributions of the transcription factors in p:
distribViz(IMP_Z,p_TFs)
```

heatI — *Generates an heatmap visualization of the mean Importance Index of pairs of transcription factors.*

Description

A square matrix (TFs x TFs) is built. The element (i,j) of such matrix contains the mean Importance Index (II) of the couple of transcription factors (TFi, TFj). This is represented in a heatmap visualization, where the scale color from blue to white indicates low mean importance of the couple and from white to red indicates high mean importance of the couple.

Usage

```
heatI(TFs, I)
```

Arguments

TFs	a string vector with names of transcription factors.
I	a vector of mean II of pairs of transcription factors in TFs.

Value

The Importance index of pairs of Transcription Factors as a heatmap

Examples

```
# Load p_TFs and I_c_2 from the data_man collection of datasets:
data('data_man')

# Heatmap visualization of the mean importances of transcription factors in p
# and their combinations in two elements:
heatI(p_TFs, I_c_2)
```

IComp *Computes the Importance Index of a transcription factor in a set of association rules.*

Description

Given an association rule and a transcription factor TFi, it is evaluated the contribution of TFi in the rule for the prediction of the presence of the item in the right-hand-side of the rule. Since this contribution is evaluated based on the variations of support and confidence of the rule, the user can visualize such variations by setting the parameter figures = TRUE.

Usage

```
IComp(TFi, rules_TF, rules_noTF, figures)
```

Arguments

TFi	string or string vector: the transcription factor (or combination of transcription factors) whose importance distribution is evaluated.
rules_TF	a set of rules in which TFi is present.
rules_noTF	a set of rules obtained from rules_TF removing the transcription factor (or combination of transcription factors) in TFi (obtained with the function rulesTF0).
figures	logical; if figures = TRUE, graphics with support and confidence distributions of the rules in rulesTF and rulesTF0 are returned.

Value

A list of four elements: the imp element is a vector of doubles with the importances of TFi in each rule in rulesTF; the delta element of the list is a list with variations of distributions of the two measures of support and confidence. This output is used in the function IPCA for the Principal Component Analysis of such distributions. The rwi element is a data.frame with the rules in rulesTF in which the transcription factor TFi is present and the rwo element is a data.frame with rules in rwi obtained removing the transcription factor TFi. Furthermore, if the input argument figures is set to TRUE, also the plots of the distributions of support and confidence of the rules before and after removing the transcription factor TFi are provided.

Examples

```
# Load r_FOSL2 and r_noFOSL2 from the data_man collection of datasets:
data('data_man')

# The Importance Indexes of FOSL2=1 in the set of rules r_FOSL2 are given by:
IComp('FOSL2=1', r_FOSL2, r_noFOSL2, figures=TRUE)
```

IMP *Contains the mean Importance Index of each co-regulator.*

Description

Within the data_man data collection, the dataset IMP has 4 columns and 12 rows: transcription factors present in at least one association rule are listed in the fist column (TF), the second column (imp) contains the means of the Importance Index associated with each trascription factor, the standard deviations for each transcription factor are reported in the third column (sd) and finally the number of rules found for each transcription factor are transcribed in the fourth column (nrules).

Usage

```
data("data_man")
```

Format

An object of class "data.frame"

Examples

```
# IMP is found in the data_man collection of datasets:
data("data_man")
head(IMP$imp)
```

imp_FOSL2 *Contains the set of Importance Indexes of FOSL2 in a given set of rules.*

Description

Within the data_man data collection, imp_FOSL2 is a list of 4 elements, containing imp as a numeric vector in which the means of Importance Indexes are represented, delta as a data.frame in which the variations of support and confidence measures are reported, rwi as a data.frame of association rules where FOSL2 is present and rwo as a data.frame in which the same rules are considered for FOSL2=0.

Usage

```
data("data_man")
```

Format

An object of class "list"

Examples

```
# imp_FOSL2 is found in the data_man collection of datasets:
data("data_man")
head(imp_FOSL2$imp)
```

IMP_Z *Contains the Importance Index associated with each co-regulator which is present in at least one association rule.*

Description

IMP_Z is a list of 12 elements, containig the Importance Indexes for all transcription factors (p) in the set of relevant association rules extracted. It is included in the data_man data collection.

Usage

```
data("data_man")
```

Format

An object of class "list"

Examples

```
# IMP_Z is found in the data_man collection of datasets:
data("data_man")
head(IMP_Z[[1]])
```

IPCA *Principal Components Analysis for distributions of variations of support and confidence obtained removing a transcription factor from a set of rules.*

Description

The function is used for validation of the defined Importance Index. It is defined as the linear combination of variations of support and confidence measures. The Principal Component Analysis lets the user evaluate if in the 1D reference system defined by such linear combination, it is possible to describe the variability of the data.

Usage

```
IPCA(delta_list, IMP)
```

Arguments

`delta_list`	list of variations of distributions of support and confidence measures, obtained using the `IComp`.
`IMP`	the importance matrix with the mean Importance Index of every candidate co-regulator transcription factor and the number of rules in which each of them appears.

Value

Variance explained by every principal component (`summary`), scores (i.e., the coordinates) of data in delta_list in the reference system defined by the principal components (`scores`) and loadings (i.e., the coefficinets) of the linear combination that defines each principal component (`loadings`). The plots of the variability, the cumulate percentage of variance explained by each principal component and loadings of every principal component are also returned.

Examples

```
# Load IMP, DELTA and TF_Imp from the data_man collection of datasets:
data('data_man')

colnames(IMP)
TF_Imp <- data.frame(IMP$TF, IMP$imp, IMP$nrules)
i.pc <- IPCA(DELTA, TF_Imp)
names(i.pc)
```

items *Extracts items in the left-hand-side of an association rule.*

Description

The function is used in the elaboration of the left-hand-side of association rules to search for the items.

Usage

```
items(itemset)
```

Arguments

itemset	object of class rules.

Value

A string vector with the items in the left-hand-side of the rule in itemset.

Examples

```
items('{TAF1=1,EP300=1,MAX=1}')
# the output is: 'TAF1=1','EP300=1','MAX=1'
```

itemset *Builds the itemset in the left-hand-side of an association rule.*

Description

The function is used in the construction of the left-hand-side of association rules to search for a rule of interest.

Usage

```
itemset(items)
```

Arguments

items	a string vector.

Value

Itemset of the left-hand-side of a searched rule, with the items in `items`.

Examples

```
itemset(c('TAF1=1','EP300=1'))
# the output is: 'TAF1=1,EP300=1'
```

I_c_2 *Contains the mean Importance Index of pairs of transcription factors which are present in at least one association rule.*

Description

Within the data_man data collection, the dataset I_c_2 has 2 columns and 78 rows: the fist column (TF) contains the transcription factors (single or in pairs) and the second column (imp) lists the Importance Indexes associated with each trascription factor or pair of transcription factors.

Usage

```
data("data_man")
```

Format

An object of class "data.frame"

Examples

```
# I_c_2 is found in the data_man collection of datasets:
data("data_man")
head(I_c_2$imp)
```

MCF7_chr1 *Contains genomic regions in the first chromosome of the MCF-7 human breast adenocarcinoma cell line at the ranges side, and the presence indexes of transcription factors in such regions at the metadata side.*

Description

MCF7_chr1 is a Large GRanges in which metadata columns identify transcription factors and genomic coordinates of regions in the first chromosome of the MCF-7 human breast adenocarcinoma cell line are represented in the left-hand side of the GRanges, therefore each row is a different genomic region.

Usage

```
data("MCF7_chr1")
```

Format

An object of class "GRanges"

Examples

```
data("MCF7_chr1")
head(MCF7_chr1)
```

`presAbs` *Splits a set of transcription factors in 'present' or 'absent' transcription factors in a set of rules.*

Description

The function is used to find candidate transcription factor co-regulator with a given transcription factor, by finding the transcription factors that contribute to the prediction of the given trascription factor presence in a set of association rules.

Usage

```
presAbs(TFs, rules, type)
```

Arguments

`TFs`	a string vector: set of transcription factors to check their presence (TF=1) or absence (TF=0) in the set of rules.
`rules`	data.frame: rules and their quality measures.
`type`	logical parameter: if `type = TRUE`, only rules with all present transcription factors in the left-hand-side are considered (i.e., the left-hand-side of the extracted rules is for example TF1=1, TF2=1, TF3=1).

Value

A list of two string vectors: the list `pres` contains all the transcription factors in `TFs` that are present in `rules`, and the list `abs` contains all the transcription factors in `TFs` that are absent in `rules`.

Examples

```
library(GenomicRanges)
# Load r_TEAD4 from the data_man collection of datasets:
data('data_man')
# Load MCF7_chr1:
data('MCF7_chr1')

# Transcription factors present in at least one of the regions
# in the considered dataset:
c <- names(elementMetadata(MCF7_chr1))

names(presAbs(c, r_TEAD4, TRUE))

# Transcription factors present in at least one of the association rules:
p_TFs <- presAbs(c, r_TEAD4, TRUE)$pres
p_TFs
```

p_TFs *Contains co-regulators found in at least one association rule.*

Description

p_TFs is a character vector that contains the transcription factors present in at least one association rule. It is included in the data_man data collection.

Usage

```
data("data_man")
```

Format

An object of class "character"

Examples

```
# p_TFs is found in the data_man collection of datasets:
data("data_man")
p_TFs
```

rulesGen *Exctracts relevant association rules.*

Description

From the dataset in data, the function extracts a set of association rules, with a certain item in their right-hand-side. Each rule extracted has support greater than minsupp and confidence greater than minconf. The extraction is made using the apriori function implemented in the arules package. minsupp and minconf thresholds are set by the user in order to extract a limited number of most relevant association rules.

Usage

```
rulesGen(data, TF, minsupp, minconf, type)
```

Arguments

data	a GRanges object in which the metadata columns contain the Indicator of presence matrix i.e., a matrix with 1 and 0 values representing presence or absence, respectively (in case other values different from 0 are present, all of them are considered as representing presence).
TF	a string with the name of the trancription factor wanted in the right-hand-side of the extracted rules.
minsupp	an integer, the minimal support of the extracted rules.

minconf	an integer, the minimal confidence of the extracted rules.
type	a logical parameter; if type = TRUE, only rules with all present transcription factor in the left-hand-side are extracted (i.e., the left-hand-side of the extracted rules is of the type TF1=1, TF2=1, TF3=1). If type = FALSE, also rules with absent transcription factors in the left-hand-side are extracted (i.e., the left-hand-side of the extracted rules can be of the type TF1=1, TF2=0, TF3=1 or TF1=0, TF2=0, TF3=0).

Value

A data frame with the association rules extracted and their quality measures of support, confidence and lift.

See Also

apriori

Examples

```
# Load the dataset:
data('MCF7_chr1')

# To extract association rules from data, with TEAD4=1 in the right-hand-side
# and support greater than 0.005 and confidence greater than 0.62:
# r_TEAD4 <- rulesGen(data, 'TEAD4=1', minsupp=0.005, minconf=0.62,
#                     type=TRUE)

r_TEAD4 <- rulesGen(MCF7_chr1, 'TEAD4=1', 0.005, 0.62, TRUE)
```

---

rulesTF *Extracts a subset of rules that contain a certain transcription factor (or a combination of transcription factors) in their left-hand-side.*

Description

From a set of relevant association rules, only the ones containing TFi in their left-hand-side are subsetted, together with their quality measures of support, confidence and lift. The function is then used for the evaluation of the importance distribution of TFi.

Usage

```
rulesTF(TFi, rules, verbose)
```

Arguments

TFi	a string with the name of the transcription factor (or combination of transcription factors) wanted in the left-hand-side of the rules to find.
rules	a data.frame with association rules and their quality measures of support, confidence and lift.
verbose	a logical parameter. If verbose = TRUE, a warning message is reported to the user when the set of rules containing TFi is empty.

Value

A data.frame with association rules containing TFi in their left-hand-side, with their quality measures of support, confidence and lift.

Examples

```
# Load r_TEAD4 from the data_man collection of datasets:
data('data_man')

r_FOSL2 <- rulesTF('FOSL2=1', r_TEAD4, verbose=FALSE)
```

---

rulesTF0 *Substitutes the presence of a transcription factor (or a combination of transcription factors) in the left-hand-side of a set of rules, with its absence.*

Description

The function substitutes the presence of a given transcription factor TFi (or a combination of transcription factors) chosen by the user with its absence, in the subset of relevant association rules extracted with the function rulesTF. Then it searches for the obtained rules and their quality measures of support, confidence and lift in the set of most relevant associations extracted with the function rulesGen. A rule is searched in all the association rules generable from the considered dataset using the function search_rule.

Usage

```
rulesTF0(TFi, sub_rules, all_rules, data, RHS)
```

Arguments

TFi	a string, or a string vector: transcription factor (or combination of transcription factors) to remove from the set of rules.
sub_rules	a data.frame with a subset of rules containing TFi, and their quality measures of support, confidence and lift (i.e., rules from which the user wants to remove TFi).

all_rules	a data.frame with a set of all the rules and their quality measures of support, confidence and lift, to be considered for the search of the obtained rules and their quality measures.
data	a GRanges object which contains the Indicator of presence matrix i.e., a matrix with 1 and 0 values representing presence or absence, respectively (in case other values different from 0 are present, all of them are considered as representing presence).
RHS	the right-hand-side of the considered association rules.

Value

A data.frame with all the rules in the set sub_rules in which the transcription factor (or combination of transcription factors) TFi is absent, and their quality measures of support, confidence and lift.

Examples

```
# Load r_TEAD4 and r_FOSL2 from the data_man collection of datasets:
data("data_man")
# Load MCF7_chr1:
data("MCF7_chr1")


r_noFOSL2 <- rulesTF0("FOSL2=1", r_FOSL2, r_TEAD4, MCF7_chr1, "TEAD4=1")
```

---

r_FOSL2

Represents an example of rulesTF output, i.e. the subset of rules whose left-hand-sides contain FOSL2, and the correspondent quality measures.

Description

Within the data_man data collection, the dataset r_FOSL2 has 5 columns and 28 rows: the first column contains the left-hand-side of the rules (lhs), the second column is the right-hand-side of the rules (rhs), the third column reports the support measures (support), the fourth column contains the confidence measures (confidence) and the lift measures are listed in the fifth column (lift).

Usage

```
data("data_man")
```

Format

An object of class "data.frame"

Examples

```
# r_FOSL2 is found in the data_man collection of datasets:
data("data_man")
head(r_FOSL2$lhs)
```

---

r_noFOSL2 *Represents an example of rulesTF0 output, where the presence of FOSL2 was replaced with its absence.*

Description

Within the data_man data collection, the dataset r_noFOSL2 has 5 columns and 28 rows: the first column contains the left-hand-side of the rules (lhs), the second column is the right-hand-side of the rules (rhs), the third column reports the support measures (support), the fourth column contains the confidence measures (confidence) and the lift measures are listed in the fifth column (lift).

Usage

```
data("data_man")
```

Format

An object of class "data.frame"

Examples

```
# Load r_noFOSL2 from the data_man collection of datasets:
data("data_man")
head(r_noFOSL2$lhs)
```

---

r_TEAD4 *Contains the association rules for the prediction of the presence of the transcription factor TEAD4 in the considered genomic regions, i.e., with TEAD4 in the right-hand-side of the association rules.*

Description

Within the data_man data collection, the dataset r_TEAD4 has 5 columns and 28 rows: the first column contains the left-hand-side of the rules (lhs), the second column is the right-hand-side of the rules (TEAD4=1) (rhs), the third column reports the support measures (support), the fourth column contains the confidence measures (confidence) and the lift measures are listed in the fifth column (lift).

Usage

```
data("data_man")
```

Format

An object of class "data.frame"

Examples

```
# Load r_TEAD4 from the data_man collection of datasets:
data("data_man")
head(r_TEAD4$lhs)
```

---

search_rule *Finds an association rule in a dataset.*

Description

The function looks for an association rule in a dataset data with a fixed left-hand-side (LHS) and a fixed right-hand-side (RHS). The function is used in the rulesTF0.

Usage

```
search_rule(data, LHS, RHS)
```

Arguments

data	a binary matrix or data.frame in which the rule is searched
LHS	a string with the left-hand-side of the searched rule
RHS	a string with the right-hand-side of the searched rule

Value

A vector with five elements: the left-hand-side of the searched rule, the right-hand-side of the searched rule, the support, confidence and lift of the found rule.

TF_Imp *Contains the candidate co-regulators and the number of rules associated with them.*

Description

Within the data_man data collection, the dataset TF_Imp has 3 columns and 12 rows: the fist column contains the transcription factors (IMP.TF), the Importance Indexes associated with each trascription factor are listed in the second column (IMP.imp) and the third column contains the number of rules found for each transcription factor (IMP.nrules).

Usage

```
data("data_man")
```

Format

An object of class "data.frame"

Examples

```
# TF_Imp is found in the data_man collection of datasets:
data("data_man")
head(TF_Imp$IMP.TF)
```

Index

∗ **datasets**
DELTA, 2
I_c_2, 9
IMP, 5
imp_FOSL2, 6
IMP_Z, 6
MCF7_chr1, 9
p_TFs, 11
r_FOSL2, 14
r_noFOSL2, 15
r_TEAD4, 15
TF_Imp, 16

apriori, *11*, *12*

DELTA, 2
distribViz, 3

heatI, 3

I_c_2, 9
IComp, 4
IMP, 5
imp_FOSL2, 6
IMP_Z, 6
IPCA, *5*, 7
items, 8
itemset, 8

MCF7_chr1, 9

p_TFs, 11
presAbs, 10

r_FOSL2, 14
r_noFOSL2, 15
r_TEAD4, 15
rulesGen, 11, *13*
rulesTF, 12, *13*
rulesTF0, *4*, 13

search_rule, *13*, 16

TF_Imp, 16