Package 'iJRF'

January 27, 2017

Version 1.1-4

Date 2016-11-1

Title Integrative Joint Random Forest

Depends R (>= 3.0.0)

Suggests MASS

Imports ggplot2, stats

Description
Integrative framework for the simultaneous estimation of interactions from different class of data.

License GPL (>= 2)

URL https://www.r-project.org

NeedsCompilation yes

Author Francesca Petralia [aut, cre],
Pei Wang [aut],
Zhidong Tu [aut],
Won-min Song [aut],
Adele Cutler [ctb],
Leo Breiman [ctb],
Andy Liaw [ctb],
Matthew Wiener [ctb]

Maintainer Francesca Petralia <francesca.petralia@mssm.edu>

Repository CRAN

Date/Publication 2017-01-27 18:47:34

R topics documented:

iJRF 2
iJRF_Perm 3
Unweighted_Network 5

Index **7**

iJRF *Integrative Joint Random Forest*

Description

iJRF infers interactions across two different sets of genomic variables for different class of data. iJRF borrows information across multiple class of data while taking into account prior information from existing databases. As an example, iJRF can be used to infer microRNA-mRNA interactions for different data sets corresponding to different treatment conditions while taking into account information from existing microRNA-mRNA databases.

Usage

```
iJRF(X, Y, W, ntree=NULL, mtry=NULL,res.name=NULL,cov.name=NULL)
```

Arguments

X	List object containing predictors for each class, `X=list(x_1,x_2, ... )` where `x_j` is a `(M x n_j)` matrix with rows corresponding to predictors and columns to samples. Missing values are not allowed.
Y	List object containing response variables for each class, `Y=list(y_1,y_2, ... )` where `y_j` is a `(p x n_j)` matrix with rows corresponding to response variables and columns to samples. Missing values are not allowed.
W	`(M x p)` Matrix containing sampling scores based on prior information on interactions. Element `(i,j)` contains interaction score `(i -> j)`. Scores must be non-negative. Larger value of sampling score corresponds to higher likelihood of variable `i` interacting with variable `j`. Rows of `W` must be in the same order as the rows of `X`, while columns of `W` must be in the same order as the rows of `Y`.
ntree	Numeric value: number of trees. If omitted, `ntree` is set to `1000`.
mtry	Numeric value: number of predictors to be sampled at each node. If omitted, `mtry` is set to the square root of the number of predictors.
res.name	p-dimensional vector containing names of response variable.
cov.name	M-dimensional vector containing names of predictors.

Value

A matrix with `I` rows and `C + 2` columns where `I=M x p` is the total number of interactions and `C` is the number of classes. The first two columns contain variables name for each interaction while the remaining columns contain importance scores for different classes.

References

Petralia, F. et al (2017) A new method to study the change of miRNA-mRNA interactions due to environmental exposures, Submitted.

Petralia, F., Wang, P., Yang, J., and Tu Z. (2015) Integrative random forest for gene regulatory network inference. **31**(12), i197-i205.

Petralia, F., Song, W.M., Tu, Z. and Wang, P. (2016). New method for joint network analysis reveals common and different coexpression patterns among genes and proteins in breast cancer. *Journal of proteome research*, **15**(3), pp.743-754.

Some of the functions utilized are a modified version of functions contained in R package randomForest: A. Liaw and M. Wiener (2002). Classification and Regression by randomForest. *R News* **2**, 18–22.

Examples

```
# --- Generate data sets
nclasses=2               # number of data sets / classes
n1<-n2<-20               # sample size for each data sets
p<-5                    # number of response variables
M<-10                    # number of predictor variables
W<-abs(matrix(rnorm(M*p),M,p))    # generate sampling scores

Res1<-matrix(rnorm(p*n1),p,n1)       # generate response for class 1
Res2<-matrix(rnorm(p*n2),p,n2)       # generate response for class 2
Cov1<-matrix(rnorm(M*n1),M,n1)       # generate predictors for class 1
Cov2<-matrix(rnorm(M*n2),M,n2)       # generate predictors for class 2

# --- Standardize variables to mean 0 and variance 1
 Res1 <- t(apply(Res1, 1, function(x) { (x - mean(x)) / sd(x) } ))
 Res2 <- t(apply(Res2, 1, function(x) { (x - mean(x)) / sd(x) } ))

# --- Run iJRF and obtain importance score of interactions
out<-iJRF(X=list(Cov1,Cov2),Y=list(Res1,Res2),W=W)
```

`iJRF_Perm` *Derive importance scores for P permuted data sets.*

Description

This function computes importance score for `P` permuted data sets. For each permuted data set, sample labels of response variable are randomly permuted and iJRF is implemented. Resulting importance scores can be used to derive an estimate of FDR.

Usage

```
iJRF_Perm(X, Y, W, ntree=NULL, mtry=NULL,res.name=NULL,cov.name=NULL,P)
```

Arguments

X	List object containing predictors for each class, `X=list(x_1,x_2, ... )` where `x_j` is a `(M x n_j)` matrix with rows corresponding to predictors and columns to samples. Missing values are not allowed.
Y	List object containing response variables for each class, `Y=list(y_1,y_2, ... )` where `y_j` is a `(p x n_j)` matrix with rows corresponding to response variables and columns to samples. Missing values are not allowed.
W	`(M x p)` Matrix containing sampling scores based on prior information on interactions. Element `(i,j)` contains interaction score `(i -> j)`. Scores must be non-negative. Larger value of sampling score corresponds to higher likelihood of variable `i` interacting with variable `j`. Rows of `W` must be in the same order as the rows of `X`, while columns of `W` must be in the same order as the rows of `Y`.
ntree	Numeric value: number of trees. If omitted, `ntree` is set to `1000`.
mtry	Numeric value: number of predictors to be sampled at each node. If omitted, `mtry` is set to the square root of the number of predictors.
res.name	p-dimensional vector containing names of response variable.
cov.name	M-dimensional vector containing names of predictors.
P	Number of permutations.

Value

A three dimensional array (`I`, `P`, `C`) where `I=M x p` is the total number of interactions, `C` is the number of classes and `P` the total number of permutations.

References

Petralia, F. et al (2017) A new method to study the change of miRNA-mRNA interactions due to environmental exposures, Submitted.

Petralia, F., Wang, P., Yang, J., and Tu Z. (2015) Integrative random forest for gene regulatory network inference. **31**(12), i197-i205.

Petralia, F., Song, W.M., Tu, Z. and Wang, P. (2016). New method for joint network analysis reveals common and different coexpression patterns among genes and proteins in breast cancer. *Journal of proteome research*, **15**(3), pp.743-754.

Some of the functions utilized are a modified version of functions contained in R package randomForest: A. Liaw and M. Wiener (2002). Classification and Regression by randomForest. *R News* **2**, 18–22.

Examples

```
# --- Generate data sets
nclasses=2               # number of data sets / classes
n1<-n2<-20               # sample size for each data sets
p<-5                    # number of response variables
M<-10                    # number of predictor variables
W<-abs(matrix(rnorm(M*p),M,p))    # generate sampling scores
```

```
Res1<-matrix(rnorm(p*n1),p,n1)       # generate response for class 1
Res2<-matrix(rnorm(p*n2),p,n2)       # generate response for class 2
Cov1<-matrix(rnorm(M*n1),M,n1)       # generate predictors for class 1
Cov2<-matrix(rnorm(M*n2),M,n2)       # generate predictors for class 2

# --- Standardize variables to mean 0 and variance 1
 Res1 <- t(apply(Res1, 1, function(x) { (x - mean(x)) / sd(x) } ))
 Res2 <- t(apply(Res2, 1, function(x) { (x - mean(x)) / sd(x) } ))

# --- Run iJRF and obtain importance score for P permuted data sets
 out<-iJRF_Perm(X=list(Cov1,Cov2),Y=list(Res1,Res2),W=W,P=2)
```

`Unweighted_Network` *Compute permutation-based FDR of importance scores and return estimated interactions.*

Description

This function computes permutation-based FDR of importance scores and returns interactions.

Usage

```
Unweighted_Network(out.iJRF,out.perm,TH)
```

Arguments

`out.iJRF`	Output from object of class `iJRF`.
`out.perm`	Output from object of class `iJRF_Perm`.
`TH`	Threshold for FDR.

Value

List of estimated interactions.

References

Petralia, F. et al (2017) A new method to study the change of miRNA-mRNA interactions due to environmental exposures, Submitted.

Petralia, F., Wang, P., Yang, J., and Tu Z. (2015) Integrative random forest for gene regulatory network inference. **31**(12), i197-i205.

Petralia, F., Song, W.M., Tu, Z. and Wang, P. (2016). New method for joint network analysis reveals common and different coexpression patterns among genes and proteins in breast cancer. *Journal of proteome research*, **15**(3), pp.743-754.

Some of the functions utilized are a modified version of functions contained in R package randomForest: A. Liaw and M. Wiener (2002). Classification and Regression by randomForest. *R News* **2**, 18–22.

Examples

```
 # --- Generate data sets
 nclasses=2               # number of data sets / classes
 n1<-n2<-20               # sample size for each data sets
 p<-5                   # number of response variables
 M<-10                   # number of predictor variables
 W<-abs(matrix(rnorm(M*p),M,p))    # generate sampling scores

 Res1<-matrix(rnorm(p*n1),p,n1)       # generate response for class 1
 Res2<-matrix(rnorm(p*n2),p,n2)       # generate response for class 2
 Cov1<-matrix(rnorm(M*n1),M,n1)       # generate predictors for class 1
 Cov2<-matrix(rnorm(M*n2),M,n2)       # generate predictors for class 2

 # --- Standardize variables to mean 0 and variance 1
  Res1 <- t(apply(Res1, 1, function(x) { (x - mean(x)) / sd(x) } ))
  Res2 <- t(apply(Res2, 1, function(x) { (x - mean(x)) / sd(x) } ))


# --- Run iJRF and obtain importance score of interactions
out.iJRF<-iJRF(X=list(Cov1,Cov2),Y=list(Res1,Res2),W=W)

# --- Run iJRF for P permutated data sets
out.perm<-iJRF_Perm(X=list(Cov1,Cov2),Y=list(Res1,Res2),W=W,P=2)

# --- Derive final networks
final.net<-Unweighted_Network(out.iJRF,out.perm,0.001)
```

Index

iJRF, 2
iJRF_Perm, 3

Unweighted_Network, 5