Package ‘CCPROMISE’

March 28, 2017

Type Package

Title PROMISE analysis with Canonical Correlation for Two Forms of High Dimensional Genetic Data

Version 1.0.0

Author Xueyuan Cao <xueyuan.cao@stjude.org> and Stanley.pounds <stanley.pounds@stjude.org>

Maintainer Xueyuan Cao <xueyuan.cao@stjude.org>

Description Perform Canonical correlation between two forms of high dimensional genetic data, and associate the first component of each form of data with a specific biologically interesting pattern of associations with multiple endpoints. A probe level analysis is also implemented.

Depends R (>= 3.3.0), stats, methods, CCP, PROMISE, Biobase, GSEABase, utils

License GPL (>= 2)

biocViews Microarray, GeneExpression

LazyLoad yes

NeedsCompilation no

R topics documented:

CCPROMISE-package ......................................................... 2
CANN .............................................................................. 3
CCPROMISE ................................................................. 4
exmplESet ............................................................... 6
exmplGeneSet .......................................................... 7
exmplMSet .............................................................. 7
exmplPat ................................................................. 8
PrbCor ................................................................. 8
PrbPROMISE .......................................................... 9
PROMISE2 ............................................................. 11

Index 13
Description

A tool to identify genes that are correlated between two sets of genomic variables and are associated with a predefined pattern of associations with multiple endpoint variables.

Details

Package: CCPROMISE
Type: Package
Version: 0.99.3
Date: 2016-10-11
License: GPL (>=2)
LazyLoad: yes

The CCPROMISE (Canonical correlation with PROMISE analysis) is performed by calling function CCPROMISE. The two forms of genomic data such as gene expression and methylation are passed through minimal ExpressionSet; the gene annotation (defining relationship between a gene and the two forms of genomic data), phenotypic data and definition of R routines for calculating association statistics with individual endpoint variables are same as in PROMISE package. Please refer to PROMISE package for writing user defined routines.

Author(s)

Xueyuan Cao <Xueyuan.cao@stjude.org>, Stanley Pounds <stanley.pounds@stjude.org>
Maintainer: Xueyuan Cao <xueyuan.cao@stjude.org>

References


Examples

```r
## load data
data(exmplESet)
data(exmplMSet)
data(exmplGeneSet)
data(exmplPat)
```
## Perform CCPROMISE test

test<- CCPROMISE(geneSet=exmplGeneSet,
ESet=exmplESet,
MSet=exmplMSet,
promise.pattern=exmplPat,
strat.var=NULL,
prlbl=NULL,
EMlbl=c("Expr", "Methyl"),
nbperm=TRUE,
max.ntail=10,
nperms=100,
seed=13)

---

**CANN**

*Canonical Correlation of Two Sets of Genomic Data*

### Description

Compute canonical correlation between two sets of genomic data.

### Usage

```r
CANN (geneSet, Edat, Mdat, EMlbl = c("Expr", "Methyl"), phdat)
```

### Arguments

- **geneSet**: a gene set collection to annotate probes to gene
- **Edat**: data frame of the first form of genomic data, such as gene expression data with row being probes and column being subjects. The column names should match the row names `phdat`
- **Mdat**: data frame of the second form of genomic data, such as methylation data with row being probes and column being subjects. The column names should match the row names `phdat`
- **EMlbl**: label of the genomic data, default=c("Expr", "Methyl") for `Edat` and `Mdat`
- **phdat**: phenotype data with row being subjects and column being phenotype variables. The row names should match the column names of `Edat` and `Mdat`

### Details

The function performs Canonical correlation between two forms genomic data for each gene (Edat and Mdat) defined by `gann`. If a gene only has one form of genomic data, the first principal component is used; If one form of data has number of probesets exceeding the number of subjects, the first number of subjects probesets are used. The function return a list of three components. See `value` for details.
Value

The output of the function is a list of length 3 with the following components:

**CCres**
- canonical correlation result: a data frame with row for each gene and six columns (Gene: gene names; n.EMlbl[1]: number of probes of first form genomic data; n.EMlbl[2]: number of probes of second form genomic data; CanonicalCR: Canonical correlation of first components; WilksPermPval: permutation p value of Wilks’ Lambda; WilksAsymPval: p value of F-approximations of Wilks’ Lambda).

**FSTccscore**
- the first component of canonical correlation: a data frame with row for each gene, first half of columns for first component of first form genomic data and second half of columns for first component of second form genomic data.

**CCload**
- a data frame of loading (each row is for a gene, first column is gene names, second column is the probeset ids of first form genomic data separated by '|', third column is the load for each probeset in first form genomic data separated by '|', fourth column is the probeset ids of second form genomic data separated by '|', fifth column is the load for each probeset in second form genomic data separated by '|')

Author(s)

Xueyuan Cao <Xueyuan.cao@stjude.org>, Stanley Pounds <stanley.pounds@stjude.org>

References


See Also

CCPROMISE

Examples

```r
## load exmplEdat exmplMdat
data(exmplESet)
data(exmplMSet)
data(exmplGeneSet)
## Perform canonical correlation test
test1<- CANN(geneSet=exmplGeneSet, Edat=exprs(exmplESet), Mdat=exprs(exmplMSet), EMLbl=c("Expr", "Methyl"), phdat=pData(exmplESet))
```

Description

PROMISE analysis of two genomic sets with multiple phenotypes under a predefined association pattern at gene level.
Usage

CCPROMISE (geneSet, ESet, MSet, promise.pattern, strat.var = NULL, 
prlbl = NULL, EMlbl = c("Expr", "Mthyl"), nbperm = FALSE, 
max.ntail = 100, nperms = 10000, seed = 13)

Arguments

geneSet a gene set collection to annotate probes to gene
ESet an ExpressionSet class contains minimum of exprs (expression matrix) of first 
form of genomic data such as gene expression and phenoData (AnnotatedDataFrame 
of end point data). Please refer to Biobase for details on how to create such an 
ExpressionSet expression set.
MSet an ExpressionSet class of second form of genomic data such as methylation 
levels, the subject id of MSet and ESet should be exactly same
promise.pattern PROMISE pattern
strat.var stratum variable
prlbl label of the genomic data, default=c('Expr', 'Methyl') for ESet and MSet
EMlbl indicator of fast permutation using negative binomial strategy, taking two valid 
values: FALSE or TRUE. The default is FALSE.
nbperm number of sucess if nbperm = T. Further permutation will not be performed for 
gene(s) or gene set(s) which max.ntail permuted statistics are greater or equal 
to the observed statistics, The default is 100.
max.ntail number of permutation, default = 10,000
nperms initial seed of random number generator. The default is 13.
seed

Details

The function performs PROMISE analysis for two forms of genomic data in minimal expression set 
format with a prefined phenotypic pattern. It calls two external function CANN and PROMISE2

Value

The output is a list of length 4. The 4 components are as following:

PRres PROMISE result for the first component of canonical correlation between two 
forms of genomic data. individual genes’ test statistics and p-values for each 
individual endpoint and PROMISE analysis
CCres result of canonical correlation analysis with six columns: Gene: Gene names; 
n.EMlbl[1]: number of probe set in the first form data; n.EMlbl[2]: number of 
probe set in the second form data; CanonicalCR: Canonical correlation of first 
components; WilksPermPval: permuatation p value of Wilks’ Lambda; Wilks-
AsymPval: p value of F-approximations of Wilks’ Lambda.
FSTccscore loads of first component of canonical correlation: a data frame of loading (each 
row is for a gene, first column is gene names, second column is the probeset 
ids of first form genomic data seperated by ’|’, third column is the load for 
each probeset in first form genomic data seperated by ’|’, fourth column is the 
probeset ids of second form genomic data seperated by ’|’, fifth column is the 
load for each probeset in second form genomic data seperated by ’|’)


**CCload**

a data frame of loading (each row is for a gene, first column is gene names, second column is the probeset ids of first form genomic data separated by ’|’, third column is the load for each probeset in first form genomic data separated by ’|’, fourth column is the probeset ids of second form genomic data separated by ’|’, fifth column is the load for each probeset in second form genomic data separated by ’|’)

**Author(s)**

Xueyuan Cao <Xueyuan.cao@stjude.org>, Stanley Pounds <stanley.pounds@stjude.org>

**References**


**See Also**

CANN PROMISE2

**Examples**

```r
## load data
data(exmplESet)
data(exmplMSet)
data(exmplGeneSet)
data(exmplPat)

## Perform canonical correlation test
test<- CCPROMISE(geneSet=exmplGeneSet,
                 ESet=exmplESet,
                 MSet=exmplMSet,
                 promise.pattern=exmplPat,
                 strat.var=NULL,
                 prlbl=NULL,
                 EMlbl=c("Expr", "Methyl"),
                 nbperm=FALSE,
                 max.ntail=10,
                 nperms=100,
                 seed=13)
```

---

**exmplESet**  

**Example of Conceptual Expression Set**

**Description**

An ExpressionSet class contains minimum of exprs (expression matrix) of gene expression and phenoData (AnnotatedDataFrame of endpoint data).

**Usage**

data(exmplESet)
exmplGeneSet

Value

an example ExpressionSet contains conceptual data of 105 expression features measured by U133A array for 151 subjects. The phenotype data has 8 columns for the same 151 subjects.

Example of Conceptual Gene Annotation

Description

An conceptual exemple of gene set collection to annotate both form of genomic data to genes. The gene names can be extracted by method of setName() and probe ids can be extracted by method of geneIds()

Usage

data(exmplGeneSet)

Value

a conceptual gene set collection of 10 genes with 319 unique U133A expression probe ids or Infinium HumanMethylation450 probe ids.

exmplMSet

Example of Conceptual Methylation Set

Description

an conceptual ExpressionSet class contains minimum of exprs (matrix) of DNA methylation and phenoData (AnnotatedDataFrame of end point data).

Usage

data(exmplMSet)

Value

an conceptual example ExpressionSet of 735 DNA methylation probe ids for 151 subjects. The phenotype data has 8 columns for the same 151 subjects
exmplPat  

**Example of Conceptual Phenotype Pattern Definition Set**

**Description**

An conceptual example of phenotype pattern definition set with three columns: stat.coef, stat.func, and endpt.vars; It defines an association pattern for three phenotypes.

**Usage**

data(exmplPat)

**Value**
a data frame

---

PrbCor  

**Probe Level Correlation of Two Sets of Genomic Data**

**Description**

Compute Spearman correlation of all probe combination between two sets of genomic data within a gene.

**Usage**

PrbCor (geneSet, Edat, Mdat, EMlbl = c("Expr", "Methyl"), phdat, pcut = 0.05)

**Arguments**

- `geneSet`: a gene set collection to annotate probes to gene
- `Edat`: data frame of the first form of genomic data, such as gene expression data with row being probes and column being subjects. The column names should match the row names `phdat`
- `Mdat`: data frame of the second form of genomic data, such as methylation data with row being probes and column being subjects. The column names should match the row names `phdat`
- `EMlbl`: label of the genomic data, default=c("Expr", "Methyl") for `Edat` and `Mdat`
- `phdat`: phenotype data with row being subjects and column being phenotype variables. The row names should match the column names of `Edat` and `Mdat`
- `pcut`: p value cutoff to eliminate probe pairs that are not significantly correlated. Default is 0.05

**Details**

The function performs Spearman correlation for all probe pairs between two forms genomic data within each gene (Edat and Mdat) defined by `gann`. If a gene only has one form of genomic data, the other form is coded as NA. The function return a list of two components. See `value` for details.
**Value**

The output of the function is a list of length 2. The 2 components are as following:

- **res**: spearman correlation result: a data frame with row for each probe pair with correlation p value < pcut and five columns; Gene: Gene names; EMlbl[1]: probe id in the first form data; EMlbl[2]: probe id in the second form data; Spearman.r.stat: Spearman r statistics; Spearman.p: Spearman p value.

- **gen**: Probe level data: a data frame with row for each probe pairs, first half of columns for first form genomic data and second half of columns for second form genomic data with sign reflecting the correlation of the probe pair.

**Author(s)**

Xueyuan Cao <Xueyuan.cao@stjude.org>, Stanley Pounds <stanley.pounds@stjude.org>

**See Also**

CCPROMISE

**Examples**

```r
## load exmplPhDat exmplEdat exmplMdat
data(exmplESet)
data(exmplMSet)
data(exmplGeneSet)
## Perform canonical correlation test
test1<- PrbCor(geneSet=exmplGeneSet,
    Edat=exprs(exmplESet),
    Mdat=exprs(exmplMSet),
    EMlbl=c("Expr", "Methyl"),
    phdat=pData(exmplESet))
```

---

**PrbPROMISE**

**PROMISE Analysis with Two Forms of Genomic Data at Probe Level**

**Description**

PROMISE analysis of two genomic sets with multiple phenotypes under a predefined association pattern at probe level.

**Usage**

```r
PrbPROMISE (geneSet, ESet, MSet, promise.pattern, strat.var = NULL,
    prilbl = NULL, EMlbl = c("Expr", "Methyl"), pcut = 0.05, nbperm = FALSE,
    max.ntail = 100, nperms = 10000, seed = 13)
```
Arguments

- geneSet: a gene set collection to annotate probes to gene
- ESet: an ExpressionSet class contains minimum of exprs (expression matrix) of first form of genomic data such as gene expression and phenoData (AnnotatedDataFrame of end point data). Please refer to Biobase for details on how to create such an ExpressionSet expression set.
- MSet: an ExpressionSet class of second form of genomic data such as methylation levels, the subject id of MSet and ESet should be exactly same
- promise.pattern: PROMISE pattern
- strat.var: stratum variable
- prlbl: labels
- EMlbl: label of the genomic data, default=c('Expr', 'Methyl') for ESet and MSet
- pcut: p value cutoff to eliminate probe pairs that are not significantly correlated. Default is 0.05
- nbperm: indicator of fast permutation using negative binomial strategy, taking two valid values: FALSE or TRUE. The default is FALSE.
- max.ntail: number of success if nbperm = T. Further permutation will not be performed for gene(s) or gene set(s) which max.ntail permuted statistics are greater or equal to the observed statistics. The default is 100.
- nperms: number of permutation, default = 10,000
- seed: initial seed of random number generator. The default is 13.

Details

The function performs PROMISE analysis for two forms of genomic data in minimal expression set format with a predefined phenotypic pattern. It calls two external functions PrbCor and PROMISE2.

Value

The output of the function is a list of length 2. The 2 components are as following:

- PRres: PROMISE result for the first component of canonical correlation between two forms of genomic data. Individual genes’ test statistics and p-values for each individual endpoint and PROMISE analysis
- CORres: result of spearman correlation analysis of probe pairs within a gene with five columns: Gene: Gene names; EMlbl[1]: probe id in the first form data; EMlbl[2]: probe id in the second form data; Spearman.rstat: Spearman r statistics; Spearman.p: Spearman p value.

Author(s)

Xueyuan Cao <xueyuan.cao@stjude.org>, Stanley Pounds <stanley.pounds@stjude.org>

See Also

PrbCor PROMISE2
**Examples**

```r
## load data
data(exmplESet)
data(exmplMSet)
data(exmplGeneSet)
data(exmplPat)
## Perform probe level PROMISE analysis
test<-PrbPROMISE(geneSet=exmplGeneSet,
                  ESet=exmplESet,
                  MSet=exmplMSet,
                  promise.pattern=exmplPat,
                  strat.var=NULL,
                  prlbl=c("LC50", "MRD22", "EFS", "PR3"),
                  EMlbl=c("Expr", "Methyl"),
                  nbperm=TRUE,
                  max.ntail=10,
                  nperms=100,
                  seed=13)
```

---

**PROMISE2**  
**PROMISE Analysis of Two Genomic Sets**

**Description**

PROMISE analysis of two genomic sets with multiple phenotypes.

**Usage**

```r
PROMISE2 (exprSet, exprSet2, geneSet = NULL, promise.pattern,
         strat.var = NULL, nbperm = FALSE, max.ntail = 100, nperms = 10000,
         seed = 13)
```

**Arguments**

- `exprSet`: expression set of first genomic data
- `exprSet2`: expression set of second genomic data
- `geneSet`: geneSet should be NULL.
- `promise.pattern`: PROMISE pattern
- `strat.var`: stratum variable
- `nbperm`: indicator of fast permutation using negative binomial strategy, taking two valid values: FALSE or TRUE. The default is FALSE.
- `max.ntail`: number of sucess if nbperm = T. Further permutation will not be performed for gene(s) or gene set(s) which max.ntail permutated statistics are greater or equal to the observed statistics, The default is 100.
- `nperms`: number of permutation, default = 10,000
- `seed`: random seed, default = 13
Details

The function performs PROMISE analysis for two set genomic data with a predefined phenotypic pattern. It is an intermediate function called by CCPROMISE to perform PROMISE analysis with canonical correlation.

Value

The output of the function is a list of length 2. The 2 components are as following:

- `generes`: individual genes' test statistics and p-values for each individual endpoint and PROMISE analysis.
- `setres`: Gene set level analysis is not implemented with value `NULL`.

Author(s)

Xueyuan Cao <xueyuan.cao@stjude.org>, Stanley Pounds <stanley.pounds@stjude.org>

See Also

CCPROMISE

Examples

```r
## load data
data(exmplESet)
data(exmplMSet)
data(exmplGeneSet)
data(exmplPat)

## Perform canonical correlation test
test<- PROMISE2(exmplESet[1:10],
                 exmplMSet[1:10],
                 promise.pattern=exmplPat,
                 strat.var=NULL,
                 nbperm=FALSE,
                 max.ntail=10,
                 nperms=100,
                 seed=13)
```
Index

*Topic misc
  exmplESet, 6
  exmplGeneSet, 7
  exmplMSet, 7
  exmplPat, 8
*Topic package
  CCPROMISE-package, 2

CANN, 3, 6
CCPROMISE, 4, 4, 9, 12
CCPROMISE-package, 2

  exmplESet, 6
  exmplGeneSet, 7
  exmplMSet, 7
  exmplPat, 8

PrbCor, 8, 10
PrbPROMISE, 9
PROMISE2, 6, 10, 11