Package 'jackstraw'

December 29, 2015

Type Package

Title Statistical Inference of Variables Driving Systematic Variation

Version 1.1

Date 2015-12-26

Author Neo Christopher Chung <nchchung@gmail.com>, Wei Hao <whao@princeton.edu>, John D. Storey <jstorey@princeton.edu>

Maintainer Neo Christopher Chung <nchchung@gmail.com>

Description Significance test for association between variables and their estimated latent variables. Latent variables may be estimated by principal component analysis (PCA), logistic factor analysis (LFA), and other techniques.

Imports corpcor, lfa, stats

Suggests parallel

License GPL-2

RoxygenNote 5.0.0

NeedsCompilation no

Repository CRAN

Date/Publication 2015-12-29 22:31:14

R topics documented:

dev.R *Compute Deviance for Logistic Factors*

Description

This function computes deviance between the full model and the null (intercept-only) model. It uses built-in R functions, namely `glm`; slow but no C++ dependencies. Make sure that `LFr1` and `LFr0` do not have intercept terms.

Usage

```
dev.R(dat, LFr1, LFr0 = NULL, p = FALSE)
```

Arguments

`dat`	a matrix with `m` rows and `n` columns.
`LFr1`	alternative logistic factors (an output from lfa or lfa.corpcor)
`LFr0`	null logistic factors (an output from lfa or lfa.corpcor)
`p`	estimate p-values (by default, "FALSE")

Value

When p=FALSE (by default), `dev.R` returns a vector of `m` deviances.

When p=TRUE, a list consisting of

`dev`	the `m` deviances
`p.value`	the `m` p-values based on a chisq distribution

Author(s)

Neo Christopher Chung <nchchung@gmail.com>

jackstraw *Non-Parametric Jackstraw (Wrapper)*

Description

Estimates statistical significance of association between variables and their latent variables (LVs).

Usage

```
jackstraw(dat, method = "PCA", FUN = NULL, r = NULL, ...)
```

Arguments

dat	a data matrix with m rows as variables and n columns as observations.
method	a LV estimation method (by default, "PCA"). Use an optional argument FUN to specify a custom method.
FUN	optionally, provide a specfic function to estimate LVs. Must output r estimated LVs in a n*r matrix.
r	a number of significant latent variables.
...	optional arguemtns passed along to a specific jackstraw function.

Details

This is a wrapper for a few different functions using the jackstraw method. Overall, it computes m p-values of association between m variables and their LVs. Its resampling strategy accounts for the over-fitting characteristics due to direct computation of LVs from the observed data and protects against an anti-conservative bias.

For advanced use, one may consider computing association between variables and a subset of r estimated LVs. For example, when there may be r=3 significant PCs, a user can carry out significance tests for the top two PCs (while adjusting for the third PC), by specifying r1=c(1,2) and r=3.

Please take a careful look at your data and use appropriate graphical and statistical criteria to determine a number of interesting/significant LVs, r. It is assumed that r latent variables account for systematic variation in the data.

For advanced usage, see jackstraw.PCA, jackstraw.LFA, and jackstraw.FUN.

If s is not supplied, s is set to about 10% of m variables. If B is not supplied, B is set to m*10/s.

Value

jackstraw returns a list consisting of

p.value	m p-values of association tests between variables and their principal components
obs.stat	m observed F-test statistics
null.stat	s*B null F-test statistics

Optional Arguments (see linked functions)

s a number of "synthetic" null variables. Out of m variables, s variables are independently permuted.

B a number of resampling iterations.

r1 a numeric vector of latent variables (e.g., PCs) of interest. Not appropriate for all methods or functions.

covariate a model matrix of covariates with n observations. Must include an intercept in the first column. Not appropriate for all methods and functions.

verbose a logical specifying to print the computational progress. By default, FALSE.

seed a seed for the random number generator.

Author(s)

Neo Christopher Chung <nchchung@gmail.com>

References

Chung and Storey (2013) Statistical significance of variables driving systematic variation in high-dimensional data Bioinformatics, 31(4): 545-554 http://bioinformatics.oxfordjournals.org/content/31/4/545

See Also

permutationPA jackstraw.PCA jackstraw.LFA jackstraw.FUN

Examples

```
set.seed(1234)
## simulate data from a latent variable model: Y = BL + E
B = c(rep(1,50),rep(-1,50), rep(0,900))
L = rnorm(20)
E = matrix(rnorm(1000*20), nrow=1000)
dat = B %*% t(L) + E
dat = t(scale(t(dat), center=TRUE, scale=TRUE))

## apply the jackstraw
out = jackstraw(dat, r=1, method="PCA")

## Use optional arguments
## For example, set s and B for a balance between speed of the algorithm and accuracy of p-values
## Not run:
out = jackstraw(dat, r=1, s=10, B=1000, seed=5678)

## End(Not run)
```

---

jackstraw.FUN *Non-Parametric Jackstraw for a Custom Function*

Description

Estimates statistical significance of association between variables and their latent variables, estimated using a custom function.

Usage

```
jackstraw.FUN(dat, FUN, r = NULL, r1 = NULL, s = NULL, B = NULL,
  covariate = NULL, compute.obs = TRUE, compute.null = TRUE,
  compute.p = TRUE, verbose = TRUE, seed = NULL)
```

Arguments

dat	a data matrix with m rows as variables and n columns as observations.
FUN	optionally, provide a specfic function to estimate LVs. Must output r estimated LVs in a n*r matrix.
r	a number of significant latent variables.
r1	a numeric vector of latent variables of interest.
s	a number of "synthetic" null variables. Out of m variables, s variables are independently permuted.
B	a number of resampling iterations.
covariate	a model matrix of covariates with n observations. Must include an intercept in the first column.
compute.obs	a logical specifying to return observed statistics. By default, TRUE.
compute.null	a logical specifying to return null statistics obtained by the jackstraw method. By default, TRUE.
compute.p	a logical specifying to return p-values. By default, TRUE.
verbose	a logical specifying to print the computational progress.
seed	a seed for the random number generator.

Value

jackstraw returns a list consisting of

p.value	m p-values of association tests between variables and their principal components
obs.stat	m observed statistics
null.stat	s*B null statistics

Author(s)

Neo Christopher Chung <nchchung@gmail.com>

References

Chung and Storey (2013) Statistical Significance of Variables Driving Systematic Variation in High-Dimensional Data. arXiv:1308.6013 [stat.ME] http://arxiv.org/abs/1308.6013

See Also

jackstraw

Examples

```
set.seed(1234)
## simulate data from a latent variable model: Y = BL + E
B = c(rep(1,50),rep(-1,50), rep(0,900))
L = rnorm(20)
E = matrix(rnorm(1000*20), nrow=1000)
dat = B %*% t(L) + E
dat = t(scale(t(dat), center=TRUE, scale=TRUE))

## apply the jackstraw with the svd as a function
out = jackstraw.FUN(dat, FUN = function(x) svd(x)$v[,1,drop=FALSE], r=1, s=100, B=50)
```

`jackstraw.LFA` *Non-Parametric Jackstraw for Logistic Factor Analysis*

Description

Estimates statistical significance of association between variables and their logistic factors (LFs).

Usage

```
jackstraw.LFA(dat, FUN = function(x) lfa(x, r)[, , drop = FALSE],
  devR = FALSE, r = NULL, r1 = NULL, s = NULL, B = NULL,
  covariate = NULL, compute.obs = TRUE, compute.null = TRUE,
  compute.p = TRUE, verbose = TRUE, seed = NULL)
```

Arguments

`dat`	a genotype matrix with `m` rows as variables and `n` columns as observations.
`FUN`	a function to use for LFA (by default, it uses the lfagen package)
`devR`	use a R function to compute deviance. By default, FALSE (uses C++).
`r`	a number of significant LFs.
`r1`	a numeric vector of LFs of interest (implying you are not interested in all `r` LFs).
`s`	a number of "synthetic" null variables. Out of `m` variables, `s` variables are independently permuted.
`B`	a number of resampling iterations. There will be a total of `s*B` null statistics.
`covariate`	a data matrix of covariates with corresponding `n` observations (do not include an intercept term).
`compute.obs`	a logical specifying to return observed statistics. By default, `TRUE`.
`compute.null`	a logical specifying to return null statistics obtained by the jackstraw method. By default, `TRUE`.
`compute.p`	a logical specifying to return p-values. By default, `TRUE`.
`verbose`	a logical specifying to print the computational progress.
`seed`	a seed for the random number generator.

Details

This function uses logistic factor analysis (LFA) from Wei et al. (2014). Particularly, dev in logistic regression (the full model with r LFs vs. the intercept null model) is used to assess association.

Value

jackstraw returns a list consisting of

p.value	m p-values of association tests between variables and their LFs
obs.stat	m observed devs
null.stat	s*B null devs

Author(s)

Neo Christopher Chung <nchchung@gmail.com>

See Also

jackstraw jackstraw.FUN

Examples

```
set.seed(1234)
## Not run:
## simulate genotype data from a logistic factor model: drawing rbinom from logit(BL)
m=5000; n=100; pi0=.9
m0 = round(m*pi0)
m1 = m-round(m*pi0)
B = matrix(0, nrow=m, ncol=1)
B[1:m1,] = matrix(runif(m1*n, min=-.5, max=.5), nrow=m1, ncol=1)
L = matrix(rnorm(n), nrow=1, ncol=n)
BL = B %*% L
prob = exp(BL)/(1+exp(BL))

dat = matrix(rbinom(m*n, 2, as.numeric(prob)), m, n)

## apply the jackstraw
out = jackstraw.LFA(dat, r=2)

## End(Not run)
```

---

jackstraw.parametric *Parametric Jackstraw*

Description

Estimates statistical significance of association between variables and their latent variables, from a parametric jackstraw procedure.

Usage

```
jackstraw.parametric(dat, FUN = function(x) fast.svd(x)$v[, 1:r, drop =
  FALSE], noise = function(x) rnorm(x, mean = 0, sd = 1), r = NULL,
  r1 = NULL, s = NULL, B = NULL, covariate = NULL, verbose = TRUE,
  seed = NULL)
```

Arguments

dat	a data matrix with m rows as variables and n columns as observations.
FUN	provide a function to estimate LVs. Must output r estimated LVs in a n*r matrix.
noise	specify a parametric distribution to generate a noise term.
r	a number of significant latent variables.
r1	a numeric vector of latent variables of interest.
s	a number of "synthetic" null variables. Out of m variables, s variables are independently permuted.
B	a number of resampling iterations.
covariate	a model matrix of covariates with n observations. Must include an intercept in the first column.
verbose	a logical indicator as to whether to print the progress.
seed	a seed for the random number generator.

Details

This function estimates statistical significance of association between variables and latent variables using a parametric distribution of a noise term. A small number s of observed variables are replaced by synthetic null variables generated from a specified distribution (such as Normal(0,1)). After applying a latent variable estimation function on this newly generated matrix (with s synthetic nulls and m-s intact observed variables), F-test statistics between estimated latent variables and s synthetic nulls are called the jackstraw statistics. P-values are computed by comparing observed F-test statistics against s*B jackstraw statistics.

Note that unless you have a strong reason to use a parametric distribution, it is advised to use the non-parametric jackstraw.

Value

jackstraw.parametric returns a list consisting of

p.value	the m p-values of association tests between variables and their principal components
obs.stat	the observed F-test statistics
null.stat	the s*B null F-test statistics

Author(s)

Neo Christopher Chung <nchchung@gmail.com>

See Also

jackstraw.FUN

jackstraw

`jackstraw.PCA` *Non-Parametric Jackstraw for Principal Component Analysis (PCA)*

Description

Estimates statistical significance of association between variables and their principal components (PCs).

Usage

```
jackstraw.PCA(dat, r1 = NULL, r = NULL, s = NULL, B = NULL,
  covariate = NULL, compute.obs = TRUE, compute.null = TRUE,
  compute.p = TRUE, verbose = TRUE, seed = NULL)
```

Arguments

`dat`	a data matrix with `m` rows as variables and `n` columns as observations.
`r1`	a numeric vector of principal components of interest. Choose a subset of `r` significant PCs to be used.
`r`	a number (a positive integer) of significant principal components. See permutationPA and other methods.
`s`	a number (a positive integer) of "synthetic" null variables. Out of `m` variables, `s` variables are independently permuted.
`B`	a number (a positive integer) of resampling iterations. There will be a total of `s*B` null statistics.
`covariate`	a data matrix of covariates with corresponding `n` observations (do not include an intercept term).
`compute.obs`	a logical specifying to return observed statistics. By default, `TRUE`.
`compute.null`	a logical specifying to return null statistics obtained by the jackstraw method. By default, `TRUE`.
`compute.p`	a logical specifying to return p-values. By default, `TRUE`.
`verbose`	a logical specifying to print the computational progress.
`seed`	a numeric seed for the random number generator.

Details

This function computes m p-values of linear association between m variables and their PCs. Its resampling strategy accounts for the over-fitting characteristics due to direct computation of PCs from the observed data and protects against an anti-conservative bias.

Provide the data matrix, with m variables as rows and n observations as columns. Given that there are r significant PCs, this function tests for linear association between m varibles and their r PCs.

You could specify a subset of significant PCs that you are interested in (PC). If PC is given, then this function computes statistical significance of association between m variables and PC, while adjusting for other PCs (i.e., significant PCs that are not your interest). For example, if you want to identify variables associated with 1st and 2nd PCs, when your data contains three significant PCs, set r=3 and PC=c(1,2).

Please take a careful look at your data and use appropriate graphical and statistical criteria to determine a number of significant PCs, r. The number of significant PCs depends on the data structure and the context. In a case when you fail to specify r, it will be estimated from a permutation test (Buja and Eyuboglu, 1992) using a function permutationPA.

If s is not supplied, s is set to about 10% of m variables. If B is not supplied, B is set to m*10/s.

Value

jackstraw returns a list consisting of

p.value	m p-values of association tests between variables and their principal components
obs.stat	m observed F-test statistics
null.stat	s*B null F-test statistics

Author(s)

Neo Christopher Chung <nchchung@gmail.com>

References

Chung and Storey (2013) Statistical Significance of Variables Driving Systematic Variation in High-Dimensional Data. arXiv:1308.6013 [stat.ME] http://arxiv.org/abs/1308.6013

See Also

jackstraw jackstraw.FUN permutationPA

Examples

```
set.seed(1234)
## simulate data from a latent variable model: Y = BL + E
B = c(rep(1,50),rep(-1,50), rep(0,900))
L = rnorm(20)
E = matrix(rnorm(1000*20), nrow=1000)
dat = B %*% t(L) + E
dat = t(scale(t(dat), center=TRUE, scale=TRUE))
```

```
## apply the jackstraw
out = jackstraw.PCA(dat, r=1)

## Use optional arguments
## For example, set s and B for a balance between speed of the algorithm and accuracy of p-values
## Not run:
## out = jackstraw.PCA(dat, r=1, s=10, B=1000, seed=5678)

## End(Not run)
```

`lfa.corpcor` *Logistic Factor Analysis without C++ Dependency*

Description

Estimate populatoin structure in genome-wide genotype matrices.

Usage

```
lfa.corpcor(x, d, ltrace = FALSE)
```

Arguments

`x`	a matrix with `m` loci (rows) and `n` observations (columns).
`d`	a number of logistic factors.
`ltrace`	a logical indicator as to whether to print the progress.

Details

It performs the logistic factor analysis, similar to `lfa` function in the lfa package. This function works without C++ dependencies. However, it would be much slower, does not include any other LFA-related functions, checks, and warnings.

Value

`lfa.corpcor` returns a `n*d` matrix of `d` logistic factors. The last column is always an intercept term.

See Also

jackstraw.LFA

Examples

```
set.seed(1234)
## simulate genotype data from a logistic factor model
m=5000; n=100; pi0=.9
m0 = round(m*pi0)
m1 = m-round(m*pi0)
B = matrix(0, nrow=m, ncol=1)
B[1:m1,] = matrix(runif(m1*n, min=-.5, max=.5), nrow=m1, ncol=1)
L = matrix(rnorm(n), nrow=1, ncol=n)
BL = B %*% L
prob = exp(BL)/(1+exp(BL))

dat = matrix(rbinom(m*n, 2, as.numeric(prob)), m, n)
out = lfa.corpcor(x=dat, d=2)
```

---

permutationPA *Permutation Parallel Analysis*

Description

Estimate a number of significant principal components from a permutation test.

Usage

```
permutationPA(dat, B = 100, threshold = 0.05, verbose = TRUE,
  seed = NULL)
```

Arguments

dat	a data matrix with m rows as variables and n columns as observations.
B	a number (a positive integer) of resampling iterations.
threshold	a numeric value between 0 and 1 to threshold p-values.
verbose	a logical indicator as to whether to print the progress.
seed	a seed for the random number generator.

Details

Adopted from sva::num.sv, and based on Buja and Eyuboglu (1992)

Value

permutationPA returns

p	a list of p-values for significance of principal components
r	an estimated number of significant principal components based on thresholding p-values at threshold

References

Buja A and Eyuboglu N. (1992) Remarks on parrallel analysis. Multivariate Behavioral Research, 27(4), 509-540

Index