Package 'MDR'

July 2, 2014

Type Package

Title Detect gene-gene interactions using multifactor dimensionality reduction

Version 1.2

Date 2012-03-11

Author Stacey Winham

Maintainer Stacey Winham <stacey.winham@gmail.com>

Description Performs multifactor dimensionality reduction (MDR) to detect potential gene-gene interactions in case-control studies.

Depends lattice

License GPL-2

LazyLoad yes

Repository CRAN

Date/Publication 2012-03-12 06:05:04

NeedsCompilation no

R topics documented:

`MDR-package` *Detect gene-gene interactions using multifactor dimensionality reduction*

Description

Performs multifactor dimensionality reduction (MDR) to detect potential gene-gene interactions in case-control studies, using balanced accuracy as an evaluation measure to rank potential models. Offers both cross-validation (CV) and a three-way split as internal validation methods to prevent over-fitting, as well as permutation testing and post-hoc prediction estimates.

Details

Package:	MDR
Type:	Package
Version:	1.1
Date:	2011-07-16
License:	GPL-2
LazyLoad:	yes

~~ This is an early test release. ~~

Index

- `mdr` Performs MDR over a specified set of combinations of variables/loci
- `mdr.hr` Estimates the accuracy of an MDR model given high-risk/low-risk status
- `mdr.cv` Implements MDR with cross-validation
- `mdr.3WS` Implements MDR with a three-way split internal model validation
- `boot.error` Calculates a post-hoc bootstrap prediction estimate of classification error
- `mdr.ca.adj` Calculates a post-hoc adjusted prediction estimate of classification accuracy
- `permute.mdr` Performs a permutation test on an object of class `mdr`
- `plot` Plots the best model of an object of class `mdr`
- `summary` Summarizes a previously fit object of class `mdr`
- `predict` Predicts case-control status on new data using a previously fit object of class `mdr`

Author(s)

Stacey Winham

Maintainers: Stacey Winham <stacey.winham@gmail.com>

Alison Motsinger-Reif <alison.motsinger@gmail.com>

David Reif <reif.david@gmail.com>

References

Ritchie MD et al (2001). Multifactor-dimensionality reduction reveals high-order interactions among estrogen-metabolism genes in sporadic breast cancer. Am J Hm Genet 69(1): 138-147.

Hahn LW, Ritchie MD, Moore JH (2003). Multifactor dimensionality reduction software for detecting gene-gene and gene-environment interactions. Bioinformatics 19(3):376-82.

Velez DR et al (2007). A balanced accuracy function for epistasis modeling in imbalanced datasets using multifactor dimensionality reduction. Genet Epidemiol 31(4): 306-315.

Winham SJ and Motsinger AA (2010). The effect of retrospective sampling on estimates of prediction error for multifactor dimensionality reduction. Annals of Human Genetics.

Winham SJ and Motsinger AA (2010). A comparison of internal validation techniques for multifactor dimensionality reduction. BMC Bioinformatics.

Edwards TL et al (2010). A General Framework for Formal Tests of Interaction after Exhaustive Search Methods with Applications to MDR and MDR-PDT. PLoS One 5(2).

---

boot.error *Function to calculate a post-hoc prediction estimate of classification error adjusted for population prevalence using bootstrap resampling*

Description

After fitting an MDR object and obtaining a best model, calculate an estimate of classification error that has been adjusted for retrospective sampling and accounts for disease prevalence using a bootstrap, as implemented in Winham SJ and Motsinger-Reif AA, 2010, "The effect of retrospective sampling on estimates of prediction error for multifactor dimensionality reduction," Annals of Human Genetics.

Usage

```
boot.error(data, prev, model, hr, b, genotype = c(0, 1, 2))
```

Arguments

`data`	the dataset; an n by (p+1) matrix where the first column is the binary response vector (coded 0 or 1) and the remaining columns are the p SNP genotypes (coded numerically)
`prev`	an estimate of population prevalence (from prior studies, etc.)
`model`	a numeric vector of the final MDR model loci

`hr`	vector of binary indicators for high-risk/low-risk of the genotype combinations of the final model loci
`b`	number of bootstrap samples
`genotype`	a numeric vector of possible genotypes arising in `data`; default is c(0,1,2), but this vector can be longer or shorter depending on if more or fewer than three genotypes are possible

Details

MDR provides a prediction error estimate of the final model calculated from retrospective data. To provide a prospective prediction estimate, an accurate estimate of the population prevalence rate must be incorporated.

Value

A list containing:

`classification error estimate`
: post-hoc prediction estimate of classification error adjusted for prevalence, measured as a percentage

`classification accuracy estimate`
: post-hoc prediction estimate of classification accuracy (100-classification error) adjusted for prevalence

...

Note

When determining the high-risk/low-risk status of a genotype combination, the order of combinations uses the convention that the genotypes of the first locus vary the most, based on the function `expand.grid`. For instance, with 3 genotypes (0,1,2), a two-way interaction results in the following 9 combinations: (0,0), (1,0), (2,0), (0,1), (1,1), (2,1), (0,2), (1,2), (2,2).

Author(s)

Stacey Winham

References

Ritchie MD et al (2001). Multifactor-dimensionality reduction reveals high-order interactions among estrogen-metabolism genes in sporadic breast cancer. Am J Hm Genet 69(1): 138-147.

Winham SJ and Motsinger-Reif AA (2010). The effect of retrospective sampling on estimates of prediction error for multifactor dimensionality reduction. Annals of Human Genetics.

See Also

`mdr.cv`, `mdr.3WS`, `mdr.ca.adj`

Examples

```
#load test data
data(mdr1)

#this runs mdr with 5-fold cross-validation on a subset of the sample data, considering all pairwise combinations
fit<-mdr.cv(mdr1[1:11],K=2,cv=5)

#calculates bootstrap estimate from b=100 bootstrap samples of the sample data for the previously fit MDR object '
boot.error(mdr1,prev=0.10, model=fit$'final model', hr=fit$'high-risk/low-risk', b=100)
```

`compare` *Function for internal use only ...*

Description

This function is for internal use only; it counts the number of matches of an individual data vector with the rows of a target matrix, and is called by the function `mdr`

Usage

```
compare(mat, vec, k)
```

Arguments

`mat`	target matrix
`vec`	vector to be matched
`k`	length of `vector`

Value

scalar, the total number of matches

Author(s)

Stacey Winham

See Also

`mdr`, `mdr.cv`, `mdr.3WS`

mdr *Function to perform MDR on a dataset for a given set of loci*

Description

Determines the top x MDR models over a specified set of combinations of loci which minimize balanced accuracy (mean of sensitivity and specificity). Ideally, should be used in conjunction with an internal validation method, such as cross-validation (mdr.cv) or a three-way split (mdr.3WS).

Usage

```
mdr(split, comb, x, ratio, equal = "HR", genotype = c(0, 1, 2))
```

Arguments

split	the dataset; an n by (p+1) matrix where the first column is the binary response vector (coded 0 or 1) and the remaining columns are the p SNP genotypes (coded numerically)
comb	a matrix of SNP combinations to consider; the rows represent a given combination and the columns represent the SNP number; to consider k-way interactions, comb should have k columns.
x	the number of "best" combinations to retain
ratio	the case/control ratio threshold to ascribe high-risk/low-risk status of a genotype combination
equal	how to treat genotype combinations with case/control ratio equal to the threshold; default is "HR" for high-risk, but can also consider "LR" for low-risk
genotype	a numeric vector of possible genotypes arising in split; default is c(0,1,2), but this vector can be longer or shorter depending on if more or fewer than three genotypes are possible

Details

MDR is a non-parametric data-mining approach to variable selection designed to detect gene-gene or gene-environment interactions in case-control studies. This function uses balanced accuracy as the evaluation measure to rank potential models.

Value

a list with the MDR model fit containing:

models	a matrix of the "best" x combinations of loci from comb; each row represents a 'model'
balanced accuracy	a vector of balanced accuracies for each of the 'best models'

high-risk/low-risk
: a matrix of the high-risk/low-risk parameterizations of the genotype combinations for each of the 'best models'; each row represents a 'model' and the associated vector is an indicator of high-risk status for each genotype combination.

...

Warning

MDR is a combinatorial search approach, so considering high-order interactions can be computationally expensive.

Note

When determining the high-risk/low-risk status of a genotype combination, the order of combinations uses the convention that the genotypes of the first locus vary the most, based on the function `expand.grid`. For instance, with 3 genotypes (0,1,2), a two-way interaction results in the following 9 combinations: (0,0), (1,0), (2,0), (0,1), (1,1), (2,1), (0,2), (1,2), (2,2).

Author(s)

Stacey Winham

References

Ritchie et al (2001). Multifactor-dimensionality reduction reveals high-order interactions among estrogen-metabolism genes in sporadic breast cancer. Am J Hm Genet 69, 138-147.

Velez et al (2007). A balanced accuracy function for epistasis modeling in imbalanced datasets using multifactor dimensionality reduction. Genet Epidemiol 31, 306-315.

See Also

`mdr.cv`,`mdr.3WS`

Examples

```
#load test data
data(mdr1)

#define matrix of all two-way combinations of 15 SNPs; this 105 by 2 matrix defines the 105 combinations of two-wa
loci<-t(combn(15,2))

#this runs mdr on the sample data, considering the two-way combinations in 'loci', saving the top 5 models, and de
fit<-mdr(mdr1,loci,x=5,ratio=1)

print(fit) #view the fitted mdr object
```

`mdr.3WS` *A function to perform MDR on a dataset using the three-way split for internal validation.*

Description

Determines the best MDR model up to a specified size of interaction `K` by minimizing balanced accuracy (arithmetic mean of sensitivity and specificity), while using a three-way split internal validation method. The three-way split randomly separates the data into training, testing, and validation sets. The function `mdr.3WS` is essentially a wrapper for the function `mdr`.

Usage

```
mdr.3WS(data, K, x = NULL, proportion = NULL, ratio = NULL, equal = "HR", genotype = c(0, 1, 2))
```

Arguments

`data`	the dataset; an n by (p+1) matrix where the first column is the binary response vector (coded 0 or 1) and the remaining columns are the p SNP genotypes (coded numerically)
`K`	the highest level of interaction to consider
`x`	the number of models from the training set to retain in the testing set
`proportion`	a three-dimensional vector specifying the ratio of split proportions training:testing:validation (default is 2:2:1 denoted as c(2,2,1))
`ratio`	the case/control ratio threshold to ascribe high-risk/low-risk status of a genotype combination
`equal`	how to treat genotype combinations with case/control ratio equal to the threshold; default is "HR" for high-risk, but can also consider "LR" for low-risk
`genotype`	a numeric vector of possible genotypes arising in `data`; default is c(0,1,2), but this vector can be longer or shorter depending on if more or fewer than three genotypes are possible

Details

MDR is a non-parametric data-mining approach to variable selection designed to detect gene-gene or gene-environment interactions in case-control studies. This function uses balanced accuracy as the evaluation measure to rank potential models. An overall best model is chosen to minimize balanced accuracy, while also preventing model over-fitting with internal validation. This function uses a three-way split of the data (training set for model building, testing set for replication, and validation set for prediction) for internal validation.

Value

An object of class 'mdr', which is a list containing:

`final model` a numeric vector of the predictors included in the final model

`final model accuracy`
the balanced accuracy of the final model from the validation set

`top models` a list containing the best model (with minimum BA) for each level of interaction, from 1 to K

`top model accuracies`
a matrix containing the training, testing, and validation accuracies for each level of interaction, from 1 to K

`high-risk/low-risk`
a vector of the high-risk/low-risk parameterizations of the genotype combinations for the final model

`genotypes` the numeric vector of possible genotypes specified

`validation method`
"3WS", since a three-way split internal validation procedure was utilized

...

Warning

MDR is a combinatorial search approach, so considering high-order interactions (i.e. large values for K) can be computationally expensive.

Note

When determining the high-risk/low-risk status of a genotype combination, the order of combinations uses the convention that the genotypes of the first locus vary the most, based on the function `expand.grid`. For instance, with 3 genotypes (0,1,2), a two-way interaction results in the following 9 combinations: (0,0), (1,0), (2,0), (0,1), (1,1), (2,1), (0,2), (1,2), (2,2).

Author(s)

Stacey Winham

References

Ritchie et al (2001). Multifactor-dimensionality reduction reveals high-order interactions among estrogen-metabolism genes in sporadic breast cancer. Am J Hm Genet 69, 138-147.

Velez et al (2007). A balanced accuracy function for epistasis modeling in imbalanced datasets using multifactor dimensionality reduction. Genet Epidemiol 31, 306-315.

Winham SJ and Motsinger AA (2010). A comparison of internal validation techniques for multifactor dimensionality reduction. BMC Bioinformatics.

See Also

`mdr.cv`, `mdr`, `boot.error`, `mdr.ca.adj`, `permute.mdr`, `plot.mdr`, `predict.mdr`, `summary.mdr`

Examples

```
#load test data
data(mdr1)

fit<-mdr.3WS(data=mdr1[,1:11], K=3, x = NULL, proportion = NULL, ratio = NULL, equal = "HR", genotype = c(0, 1, 2)

print(fit) #view the fitted mdr object

summary(fit) #create summary table of best MDR model

plot(fit, data=mdr1) #create contingency plot of best MDR model; may need to expand the plot window for large valu
```

`mdr.ca.adj` *Function to calculate a post-hoc adjusted prediction estimate of classification accuracy, corrected for prospective data with previously estimated population prevalence*

Description

After fitting an object of class `'mdr'` and obtaining a best model, calculate an adjusted estimate of classification accuracy to be used for prediction that accounts for retrospective sampling and incorporates disease prevalence, as implemented in Winham and Motsinger-Reif 2010.

Usage

```
mdr.ca.adj(data, model, hr, prev, genotype = c(0, 1, 2))
```

Arguments

`data`	the dataset; an n by (p+1) matrix where the first column is the binary response vector (coded 0 or 1) and the remaining columns are the p SNP genotypes (coded numerically)
`model`	a numeric vector of the final MDR model loci
`hr`	vector of binary indicators for high-risk/low-risk of the genotype combinations of the final model loci
`prev`	an estimate of population prevalence
`genotype`	a numeric vector of possible genotypes arising in `data`; default is c(0,1,2), but this vector can be longer or shorter depending on if more or fewer than three genotypes are possible

Details

MDR provides a prediction error estimate of the final model calculated from retrospective data. To provide a prospective prediction estimate, an accurate estimate of the population prevalence rate must be incorporated.

Value

List containing:

`adjusted classification accuracy`
: post-hoc prediction estimate of classification accuracy adjusted for prevalence, measured as a percentage

`adjusted classification error`
: post-hoc prediction estimate of classification error (100-classification accuracy) adjusted for prevalence

...

Note

When determining the high-risk/low-risk status of a genotype combination, the order of combinations uses the convention that the genotypes of the first locus vary the most, based on the function `expand.grid`. For instance, with 3 genotypes (0,1,2), a two-way interaction results in the following 9 combinations: (0,0), (1,0), (2,0), (0,1), (1,1), (2,1), (0,2), (1,2), (2,2).

Author(s)

Stacey Winham

References

Ritchie MD et al (2001). Multifactor-dimensionality reduction reveals high-order interactions among estrogen-metabolism genes in sporadic breast cancer. Am J Hm Genet 69(1): 138-147.

Winham SJ and Motsinger AA (2010). The effect of retrospective sampling on estimates of prediction error for multifactor dimensionality reduction. Annals of Human Genetics.

See Also

`mdr.cv`, `mdr.3WS`, `boot.error`

Examples

```
#load test data
data(mdr1)

#this runs mdr with 5-fold cross-validation on a subset of the sample data, considering all pairwise combinations
fit<-mdr.cv(mdr1[,1:11],K=2,cv=5)

#calculates adjusted CA estimate from the sample data for the previously fit MDR object 'fit', assuming the popula
mdr.ca.adj(mdr1, model=fit$'final model', hr=fit$'high-risk/low-risk', prev=0.10)
```

`mdr.cv` *A function to perform MDR on a dataset using k-fold cross-validation for internal validation.*

Description

Determines the best MDR model up to a specified size of interaction `K` by minimizing balanced accuracy (mean of sensitivity and specificity), while using a k-fold cross-validation internal validation method. The function `mdr.cv` is essentially a wrapper for the function `mdr`.

Usage

```
mdr.cv(data, K, cv, ratio = NULL, equal = "HR", genotype = c(0, 1, 2))
```

Arguments

`data`	the dataset; an n by (p+1) matrix where the first column is the binary response vector (coded 0 or 1) and the remaining columns are the p SNP genotypes (coded numerically)
`K`	the highest level of interaction to consider
`cv`	the number of cross-validation intervals; for k-fold cross-validation, cv=k
`ratio`	the case/control ratio threshold to ascribe high-risk/low-risk status of a genotype combination
`equal`	how to treat genotype combinations with case/control ratio equal to the threshold; default is "HR" for high-risk, but can also consider "LR" for low-risk
`genotype`	a numeric vector of possible genotypes arising in `data`; default is c(0,1,2), but this vector can be longer or shorter depending on if more or fewer than three genotypes are possible

Details

MDR is a non-parametric data-mining approach to variable selection designed to detect gene-gene or gene-environment interactions in case-control studies. This function uses balanced accuracy as the evaluation measure to rank potential models. An overall best model is chosen to minimize balanced accuracy, while also preventing model over-fitting with internal validation. This function uses cv-fold cross-validation to separate the data into training and testing sets. The data is randomly separated into cv equal pieces and cv-1/cv of the data is used for training/model-building and 1/cv for testing/prediction; this procedure is repeated cv times.

Value

An object of class `'mdr'`, which is a list containing:

`final model`	a numeric vector of the predictors included in the final model
`final model accuracy`	the balanced accuracy of the final model from the validation set

`top models` a list containing the best model (with minimum BA) for each level of interaction, from 1 to `K`

`top model accuracies` a matrix containing the training, testing, and validation accuracies for each level of interaction, from 1 to `K`

`high-risk/low-risk` a vector of the high-risk/low-risk parameterizations of the genotype combinations for the final model

`genotypes` the numeric vector of possible genotypes specified

`validation method` "CV", since cross-validation was utilized for internal validation

...

Warning

MDR is a combinatorial search approach, so considering high-order interactions (i.e. large values for `K`) can be computationally expensive.

Note

When determining the high-risk/low-risk status of a genotype combination, the order of combinations uses the convention that the genotypes of the first locus vary the most, based on the function `expand.grid`. For instance, with 3 genotypes (0,1,2), a two-way interaction results in the following 9 combinations: (0,0), (1,0), (2,0), (0,1), (1,1), (2,1), (0,2), (1,2), (2,2).

Author(s)

Stacey Winham

References

Ritchie et al (2001). Multifactor-dimensionality reduction reveals high-order interactions among estrogen-metabolism genes in sporadic breast cancer. Am J Hm Genet 69, 138-147.

Hahn LW, Ritchie MD, Moore JH (2003). Multifactor dimensionality reduction software for detecting gene-gene and gene-environment interactions. Bioinformatics 19(3):376-82.

Velez et al (2007). A balanced accuracy function for epistasis modeling in imbalanced datasets using multifactor dimensionality reduction. Genet Epidemiol 31, 306-315.

Motsinger AA, Ritchie MD (2006). The effect of reduction in cross-validation intervals on the performance of multifactor dimensionality reduction. Genet Epidemiol 30(6):546-55.

See Also

`mdr.3WS`, `mdr`, `boot.error`, `mdr.ca.adj`, `permute.mdr`, `plot.mdr`, `predict.mdr`, `summary.mdr`

Examples

```
#load test data
data(mdr1)

fit<-mdr.cv(data=mdr1[,1:11], K=2, cv=5, ratio = NULL, equal = "HR", genotype = c(0, 1, 2)) #fit MDR with 5-fold c

print(fit) #view the fitted mdr object

summary(fit) #create summary table of best MDR model

plot(fit, data=mdr1) #create contingency plot of best MDR model; may need to expand the plot window for large valu
```

`mdr.hr` *Function to estimate the accuracy of an MDR model given high-risk/low-risk status of genotype combinations*

Description

Determines the balanced accuracy (mean of sensitivity and specificity) of an MDR model (specified combination of loci and high-risk/low-risk genotype combinations) which minimize balanced accuracy. Is used to determine prediction error estimates in cross-validation (`mdr.cv`).

Usage

```
mdr.hr(split, model, hr, genotype = c(0, 1, 2))
```

Arguments

`split`	the dataset; an n by (p+1) matrix where the first column is the binary response vector (coded 0 or 1) and the remaining columns are the p SNP genotypes (coded numerically)
`model`	a numeric vector of the final MDR model loci
`hr`	vector of binary indicators for high-risk/low-risk of the genotype combinations of the final model loci
`genotype`	a numeric vector of possible genotypes arising in `split`; default is c(0,1,2), but this vector can be longer or shorter depending on if more or fewer than three genotypes are possible

Details

When determining the high-risk/low-risk status of a genotype combination, the order of combinations uses the convention that the genotypes of the first locus vary the most, based on the function `expand.grid`. For instance, with 3 genotypes (0,1,2), a two-way interaction results in the following 9 combinations: (0,0), (1,0), (2,0), (0,1), (1,1), (2,1), (0,2), (1,2), (2,2).

Value

List containing:

`balanced accuracy`
: Balanced accuracy estimate (the mean of sensitivity and specificity) of the specified MDR model

...

Author(s)

Stacey Winham

References

Ritchie et al (2001). Multifactor-dimensionality reduction reveals high-order interactions among estrogen-metabolism genes in sporadic breast cancer. Am J Hm Genet 69, 138-147.

Velez et al (2007). A balanced accuracy function for epistasis modeling in imbalanced datasets using multifactor dimensionality reduction. Genet Epidemiol 31, 306-315.

See Also

mdr,mdr.cv,mdr.3WS

Examples

```
#load test data
data(mdr1)

#split data into training and testing sets
train<-mdr1[1:125,]
test<-mdr1[-(1:125),]

#define matrix of all two-way combinations of 15 SNPs; this 105 by 2 matrix defines the 105 combinations of two-wa
loci<-t(combn(15,2))

#this runs mdr on the training data, considering the two-way combinations in 'loci', saving the top model, and def
fit<-mdr(train,loci,x=1,ratio=1)

#estimate balanced accuracy given the MDR best model
acc<-mdr.hr(test,model=fit$models, hr=fit$high)

print(acc)
```

mdr1 *Sample data for MDR package for n=250, p=25*

Description

This dataset provides case/control disease status and genetic information.

Usage

```
data(mdr1)
```

Format

A simulated data frame with 250 observations on 26 variables. 'Response' is a binary vector representing case(1) or control(0) status for a disease. Variables 'SNP.1' to 'SNP.25' are numeric variables which represent genotype information (coded as 0,1,2) at 25 loci.

Details

This data was simulated with an equal number of cases and controls according to a variation on the dominant-dominant model of Neuman and Rice and represents a two-way interaction with main effects at 5 percent heritability. The true disease-causing loci are SNP.4 and SNP.9, generated with minor allele frequency 0.5. The expected balanced accuracy for this model is 66.16

The penetrance function used to generate the case/control data based on the 9 possible genotype combinations is as follows:

Genotype	BB	Bb	bb
AA	0.05	0.05	0.05
Aa	0.05	0.206	0.206
aa	0.05	0.206	0.206

References

Neuman RJ, Rice JP. (1992). TWO-LOCUS MODELS OF DISEASE. Genetic Epidemiology 9(5):347-365.

Culverhouse R, et al (2002). A perspective on epistasis: limits of models displaying no main effect. Am J Hum Genet, 70(2):461-471.

Examples

```
data(mdr1)
```

mdr2 *Sample data for MDR package for n=250, p=100*

Description

This dataset provides case/control disease status and genetic information.

Usage

```
data(mdr2)
```

Format

A simulated data frame with 250 observations on 101 variables. 'Response' is a binary vector representing case(1) or control(0) status for a disease. Variables 'SNP.1' to 'SNP.100' are numeric variables which represent genotype information (coded as 0,1,2) at 100 loci.

Details

This data was simulated with an equal number of cases and controls according to a variation on the purely-epistatic XOR model of Li and Reich and represents a two-way interaction in the absence of marginal effects at 5 percent heritability. The true disease-causing loci are SNP.4 and SNP.9, generated with minor allele frequency 0.5. The expected balanced accuracy for this model is 67.09

The penetrance function used to generate the case/control data based on the 9 possible genotype combinationsis as follows:

Genotype	BB	Bb	bb
AA	0.199	0.05	0.199
Aa	0.05	0.199	0.05
aa	0.199	0.05	0.199

References

Li W, Reich J. 2000. A complete enumeration and classification of two-locus disease models. Hum Hered 50(6):334-49.

Culverhouse R, et al (2002). A perspective on epistasis: limits of models displaying no main effect. Am J Hum Genet, 70(2):461-471.

Examples

```
data(mdr2)
```

permute.mdr *Function to perform a permutation test after fitting an MDR model*

Description

After fitting an object of class 'mdr', performs a permutation test to assess the statistical significance of the balanced accuracy evaluation measure of the 'best model'.

Usage

```
permute.mdr(accuracy, loci, N.permute, method = c("CV", "3WS", "none"), data, cv, K, x = NULL, proport
```

Arguments

accuracy	the accuracy measure reported from the MDR model fit (after fitting mdr.cv, mdr.3WS, or mdr)
loci	the identified loci from the MDR model fit with mdr.cv or mdr.3WS, or prespecified set of loci fit with mdr
N.permute	the number of data permutations to perform
method	internal validation method used to fit the model: "CV" for mdr.cv, "3WS" for mdr.3WS, "none" for mdr
data	dataset used to fit the MDR model; first column is the binary response vector and subsequent columns are numeric SNP data
cv	if method="CV", the number of cross-validation intervals
K	the maximum size of interaction to consider
x	if method="3WS", the number of models to save from the training set to be evaluated in the testing set; if NULL, default is number of total loci
proportion	if method="3WS", a vector with the ratio of data for training:testing:validation sets; if NULL, default is c(2,2,1)
ratio	case/control ratio threshold to ascribe high-risk/low-risk status of a genotype combination; if NULL, default is the ratio of cases to controls in the whole dataset
equal	how to treat genotype combinations with case/control ratio equal to the threshold; if NULL, default is "HR" for high-risk, but can also consider "LR" for low-risk
genotype	a numeric vector of possible genotypes arising in data; if NULL, default is c(0,1,2)
LRT	a logical indicating if a likelihood ratio test for significant interaction should be performed

Details

Obtains permuted datasets by permuting the response vector only, in order to preserve the LD structure within the genetic data.

Value

Returns a list with:

`Permutation P-value`
: the empirical p-value based on the permutation distribution; i.e. the proportion of permutations with balanced accuracy > `accuracy`

`Permutation Distribution`
: a vector with the top balanced accuracies from all `N.permute` permutations

`LRT P-value`
: if LRT=TRUE, the empirical p-value for a test of interaction based on the LRT distribution

`LRT Distribution`
: if LRT=TRUE, a vector with p-values for the LRT test of interaction from all `N.permute` permutations

...

Warning

MDR is a combinatorial search approach, so considering high-order interactions and a large number of permutations can be computationally expensive.

Note

When using `permute.mdr` in conjunction with `mdr.cv` and `mdr.3WS`, the full internal validation and selection procedure is repeated for each permutation. For `mdr`, permutation is only consider for the specified variable combination, so internal validation or selection are not performed within each permutation.

Author(s)

Stacey Winham

References

Ritchie MD et al (2001). Multifactor-dimensionality reduction reveals high-order interactions among estrogen-metabolism genes in sporadic breast cancer. Am J Hm Genet 69(1): 138-147.

Hahn LW, Ritchie MD, Moore JH (2003). Multifactor dimensionality reduction software for detecting gene-gene and gene-environment interactions. Bioinformatics 19(3):376-82.

Velez DR et al (2007). A balanced accuracy function for epistasis modeling in imbalanced datasets using multifactor dimensionality reduction. Genet Epidemiol 31(4): 306-315.

Motsinger-Reif AA (2008). The effect of alternative permutation testing strategies on the performance of multifactor dimensionality reduction. BMC Research Notes 1:139.

Edwards TL et al (2010). A General Framework for Formal Tests of Interaction after Exhaustive Search Methods with Applications to MDR and MDR-PDT. PLoS One 5(2).

See Also

`mdr.cv`, `mdr.3WS`, `mdr`

Examples

```
#load data
data(mdr1)

#fit an mdr object to a subset of the sample data
fit<-mdr.3WS(data=mdr1[,1:11],K=2)

####save the accuracy
acc<-fit$'final model accuracy'

###save the final model loci
loc<-fit$'final model'

####run permutation test on 10 permutations
perm<-permute.mdr(accuracy=acc, loci=loc, N.permute=10, method="3WS",data=mdr1[,1:11], K=2, LRT=TRUE)

###empirical p-value
perm$'Permutation P-value'
```

`plot.mdr` *Plotting the results of an MDR model*

Description

a method for class `'mdr'` to plot case/control counts for each factor combination of a previously fit MDR model

Usage

```
## S3 method for class 'mdr'
plot(x, data, main="", xlab="", ylab="Count", table=FALSE,...)
```

Arguments

`x`	an object of class `'mdr'`, a result of a call to either `mdr.cv` or `mdr.3WS`
`data`	data set used to fit `object`
`main`	title for the plot; default is no title
`xlab`	Label for the x-axis; default is no label
`ylab`	Label for the y-axis; default is "Count"
`table`	logical for whether a summary table of case/control counts should be produced; default is FALSE
`...`	additional arguments

Details

A call to `plot` produces a trellis-style bar chart of case and control counts for each factor combination from `object`. Cases are plotted in black and controls are plotted in white. Factor combinations considered 'high-risk' are shaded gray, as seen in the legend.

Note

Requires the package `lattice`. For models of size 3 or greater, stretch the plot window for better viewing.

Author(s)

Stacey Winham

See Also

`mdr.cv`, `mdr.3WS`, `predict.mdr`, `summary.mdr`

Examples

```
#load data
data(mdr1)

#fit mdr model to a subset of the sample data
fit<-mdr.cv(data=mdr1[,1:11], K=2, cv=5)

plot(fit, data=mdr1)
```

---

predict.mdr *MDR model predictions*

Description

method for class `'mdr'` to predict case/control status for new data using a previously fit MDR model.

Usage

```
## S3 method for class 'mdr'
predict(object, new.data,...)
```

Arguments

`object`	an object of class `'mdr'`, a result of a call to either `mdr.cv` or `mdr.3WS`
`new.data`	a new data set with the same original variables in the same order, without the response
`...`	additional arguments

Value

a vector with predicted binary status ...

Author(s)

Stacey Winham

See Also

mdr.cv, mdr.3WS, summary.mdr, plot.mdr

Examples

```
#load data
data(mdr1)

#fit mdr model to a subset of the sample data
fit<-mdr.cv(data=mdr1[,1:11], K=2, cv=5)

#create 'new' data from which to predict
new<-mdr1[,2:11] #same predictor variables, without the response

predict(fit, new.data=new) #predict case/control status for this 'new' data
```

---

summary.mdr *Summarizing the results of an MDR model*

Description

summary method for class 'mdr', after fitting with mdr.cv or mdr.3WS

Usage

```
## S3 method for class 'mdr'
summary(object,...)
```

Arguments

object	an object of class 'mdr', a result of a call to either mdr.cv or mdr.3WS
...	additional arguments

Value

a table, with columns for level of interaction, bests MDR models for each level (including overall best model), and accuracy results. Accuracy results depend on the validation method for `object`.

`Level` level of interaction

`Best Models` best MDR by level

`Classification Accuracy`
average classification accuracy (percent) calculated from the training sets; for `mdr.cv`

`Prediction Accuracy`
average prediction accuracy (percent) calculated from the testing sets; for `mdr.cv`

`Cross-Validation Consistency`
the number for times a model was chosen as 'best' out of k, for k-fold cross-validation; for `mdr.cv`

`Training Accuracy`
classification accuracy (percent) calculated from the training set; for `mdr.3WS`

`Testing Accuracy`
classification accuracy (percent) calculated from the testing set; for `mdr.3WS`

`Validation Accuracy`
classification accuracy (percent) calculated from the validation set; for `mdr.3WS`

...

Author(s)

Stacey Winham

See Also

`mdr.cv`, `mdr.3WS`, `predict.mdr`, `plot.mdr`

Examples

```
#load test data
data(mdr1) #consider a subset with the response and the first 10 predictors

fit1<-mdr.3WS(data=mdr1[,1:11],K=2) #fit mdr model with 3WS
summary(fit1) #summarizes results of the fit

fit2<-mdr.cv(data=mdr1[,1:11],K=2,cv=5) #fit mdr model with 5-fold CV
summary(fit2)
```

test *Sample data for MDR package for n=1000, p=5000*

Description

This dataset provides case/control disease status and genetic information.

Usage

```
data(test)
```

Format

A simulated data frame with 1000 observations on 5001 variables. 'Response' is a binary vector representing case(1) or control(0) status for a disease. Variables 'SNP.1' to 'SNP.5000' are numeric variables which represent genotype information (coded as 0,1,2) at 5000 loci.

Details

This data was simulated with a larger number of samples and genetic predictor variables than `mdr1` and `mdr2` as an example for a larger association study. Can be used as part of a training-testing framework to assess models built with `train`

See Also

`train`

Examples

```
data(test)
```

train *Sample data for MDR package for n=1000, p=5000*

Description

This dataset provides case/control disease status and genetic information.

Usage

```
data(train)
```

Format

A simulated data frame with 1000 observations on 5001 variables. 'Response' is a binary vector representing case(1) or control(0) status for a disease. Variables 'SNP.1' to 'SNP.5000' are numeric variables which represent genotype information (coded as 0,1,2) at 5000 loci.

Details

This data was simulated with a larger number of samples and genetic predictor variables than `mdr1` and `mdr2` as an example for a larger association study. Can be used as part of a training-testing framework to build models to assess with `test`

See Also

`test`

Examples

```
data(train)
```

Index