Package 'zfa'

June 27, 2018

Type Package

Title Zoom-Focus Algorithm

Version 3.0

Date 2018-06-27

Author Haoyi Weng & Maggie Haitian Wang

Maintainer Haoyi Weng <hyweng@link.cuhk.edu.hk>

Description Performs both Zoom-Focus Algorithm (ZFA) and fuzzy ZFA to optimize and visualize testing regions for rare variant association tests in exome sequencing data.

Depends R (>= 3.1.0)

License GPL

LazyData TRUE

RoxygenNote 6.0.1

Imports SKAT (>= 1.1.2),
ggplot2 (>= 2.2.1)

R topics documented:

`fuzzy.zfa` *Fuzzy Zoom-Focus Algorithm*

Description

The function automatically locates the optimal testing region of desired rare variant method, which requires both genotype and phenotype data as input. It is applicable to exome sequenced genes with vare variants, which calls existing rare variant method functions to conduct rare variant association test.

The fuzzy Zoom-Focus algorithm (ZFA) takes two main steps, fuzzy Zooming and Focusing, to search for an optimal testing region of a given gene of any size. Given a genomic region (e.g. a gene or functional unit) of arbitrary size, the fuzzy Zooming step first performs a global search within the gene to locate an optimal zoomed region among all fuzzy partitons at different orders. The boundaries of the zoomed region are then refined in the next step Focusing by by extending both lower and upper bound.

Usage

```
fuzzy.zfa(data, y, fast.path = TRUE, filter.pval = 0.001, test = c("SKAT",
  "SKATO", "burden", "wtest"))
```

Arguments

`data`	a data frame or numeric matrix, usually a gene. Genotypes should be coded as 0,1 or 2.
`y`	a numeric vector with two levels. Phenotype values are coded as 0 or 1.
`fast.path`	a logical value indicating whether or not to use the fast-Zoom approach. The fast-Zoom performs a binary search instead of exhaustive search, such that at each partition order, the region is divided into two parts, only the part with smaller p-value is continued for the next level search. Default = TRUE.
`filter.pval`	a p-value threshold to select zoomed region for conducting focusing step. When specified, only zoomed region with p-value smaller than the threshold will be passed to focusing step. Default=0.01. Set filter.pval=NULL for conducting focusing step in any case.
`test`	a character to choose the rare variant method that combines with the fuzzy Zoom-Focus Algorithm.If test = "SKAT", the SKAT of variance component test is applied. If test = "SKATO", the SKAT-O of combination method test is applied. If test = "burden", the weighted burden test is applied. If test = "wtest", the W-test of burden test category is applied.

Details

The algorithm automatically selects an optimized testing region within a given gene, which enables convenient combination of various rare variant methods and, yields power improvement. Current version includes 4 existing rare variant tests. The SKAT, SKAT-O and burden test are called from `'SKAT'` package, and the W-test Collapsing method is self-contained.

Value

The "fuzzy.zfa" function returns a table with following components:

lower.bound	Lower bound of the optimized region
upper.bound	Upper bound of the optimized region
opt.region.size	Optimized region size
odds.ratio	Regional odds ratio of collapsed variants within optimal region, see odds.ratio for more details
corrected.pvalue	Bonferroni corrected p-value of optimal region

Note that fuzzy ZFA assumes that the input variants are arranged by chromosome positions, such as a functional gene. If variants from non-adjacent genomic regions are input as one data, the zfa will still treat them as adjacent. In this case, the user should be careful in interpreting the results: when an optimized region consists of distant variants, the region may not be biologically meaningful.

Author(s)

Haoyi Weng & Maggie Wang

References

M. H. Wang., H. Weng., et al. (2017) A Zoom-Focus algorithm (ZFA) to locate the optimal testing region for rare variant association tests. Bioinformatics.

M. H. Wang., R. Sun., et al. (2016) A fast and powerful W-test for pairwise epistasis testing. Nucleic Acids Research.doi:10.1093/nar/gkw347.

R. Sun., H. Weng., et al. (2016) A W-test collapsing method for rare-variant association testing in exome sequencing data. Genetic Epidemiology, 40(7): 591-596.

Wu, M. C., Lee, S., et al. (2011) Rare Variant Association Testing for Sequencing Data Using the Sequence Kernel Association Test (SKAT). American Journal of Human Genetics, 89, 82-93.

Lee, S., Emond, M.J., et al. (2012) Optimal unified approach for rare variant association testing with application to small sample case-control whole-exome sequencing studies. American Journal of Human Genetics, 91, 224-237.

Lee, S., Fuchsberger, C., et al. (2015) An efficient resampling method for calibrating single and gene-based rare variant association analysis in case-control studies. Biostatistics, kxv033.

See Also

test.variants, odds.ratio, zfa.plot

Examples

```
data(fuzzy.zfa.example)
attach(fuzzy.zfa.example)

# fast-zoom with wtest, then pass zoomed region to focusing step, and output optimization results
zfaResult1<-fuzzy.zfa(Z,y,fast.path = TRUE,filter.pval=NULL,test = "wtest")

# fuzzy zooming with wtest, only if zoomed region has p-value<0.01 will be passed to focusing step
zfaResult2<-fuzzy.zfa(Z,y,fast.path = FALSE,filter.pval=0.01,test = "wtest")
```

fuzzy.zfa.example *Example data for fuzzy ZFA*

Description

Example data for "fuzzy.zfa" function.

Usage

```
data(fuzzy.zfa.example)
```

Format

The fuzzy.zfa.example contains the following objects:

Z

a numeric genotype matrix of 596 individuals and 93 rare variants. Each row represents a different individual, and each column represents a different rare genetic variant. Genotypes are coded as 0,1 or 2.

y

a numeric vector of binary phenotypes, among which there are 298 cases and 298 controls.

See Also

fuzzy.zfa, test.variants, odds.ratio, zfa.plot

genomic.control *Genomic inflation factor for population stratification control*

Description

This function computes the genomic inflation factor to measure and control the potential inflation of p-values due to population stratification.

Usage

```
genomic.control(pvalue)
```

Arguments

pvalue	a vector of ZFA p-values in genome-wide association studies

Details

To control the potential genomic inflation due to population stratification, we follow Babron M C et. al (2012) and Liu Q et.al (2013) to measure and correct the inflation of p-values. In the absence of stratification, the observed p-values are uniformly distributed, and the -2*log(p-values) follows a 2-df chi-squared distribution. Similar to the Genomic Control lambda, a genomic inflation factor lambda is employed, measuring on the log-transformed ZFA p-values, where `lambda = -2median(log(p-values))/1.`
After computing lambda, the p-values could be adjusted by formula `exp(log(p-values)/lambda)`.

Note that the ZFA p-value has already been adjusted by Bonferroni correction. In this case, we are likely to obtain a bimodal histogram of p-values with two peaks at 0 and 1, which is due to the multiple correction. Such p-values at 1 are apparently not uniformly distributed, and are therefore filtered when calculating lambda.

Value

The function returns a list with two components:

`lambda`	Genomic inflation factor, which is measured as the rartio of median of -2*log(observed p-value) over the expected median of a 2df chi-squared distribution.
`pval.corrected`	Corrected p-values adjusting by lambda.

References

Liu Q, Nicolae D L, Chen L S. Marbled Inflation From Population Structure in Gene Based Association Studies With Rare Variants[J]. Genetic epidemiology, 2013, 37(3): 286-292.

Babron M C, De Tayrac M, Rutledge D N, et al. Rare and low frequency variant stratification in the UK population: description and impact on association tests[J]. PloS one, 2012, 7(10): e46519.

Examples

```
# run genomic control
set.seed(1121)
pvalue<-rnorm(10000, mean = 0.4, sd=0.1)
out<-genomic.control(pvalue)

# lambda
lambda<-out$lambda

# corrected p-values adjusting by lambda
pvalue.corrected<-out$pval.corrected
```

---

odds.ratio *Odds ratio of single or collapsed variants*

Description

This function simply calculates either odds ratio of each single variant or odds ratio of collapsed variants within input data.

Usage

```
odds.ratio(data, y, collapse = FALSE)
```

Arguments

data	a data frame or numeric matrix, usually a gene. Genotypes should be coded as 0,1 or 2.
y	a numeric vector with two levels. Phenotype values are coded as 0 or 1.
collapse	a logical value indicating whether or not to calculate odds ratio of each single variant. Default = FALSE for single variant calculation, or TRUE for collapsed variants calculation.

Details

When "collapse" is true, a regional odds ratio will be used to represent the overall effect size and direction of variants within a genomic region. It is calculated by summing up all the allele counts to form a new contingency table, and then calcuating the regional odds ratio from the combined contingency table.

See Also

fuzzy.zfa, test.variants, zfa.plot

Examples

```
data(fuzzy.zfa.example)
attach(fuzzy.zfa.example)

# Single variant calculation
ORSingle<-odds.ratio(Z,y,collapse=FALSE)

# Collapsed variants calculation
ORCollapse<-odds.ratio(Z,y,collapse=TRUE)
```

---

test.variants *Test of single or collapsed variants*

Description

This function performs either single-variant association tests for desired method or region-based rare variant test of collapsed variants. Four alternative methods are supported in the current version.

Usage

```
test.variants(data, y, collapse = FALSE, test = c("SKAT", "SKATO", "burden",
  "wtest"))
```

Arguments

data	a data frame or numeric matrix, usually a gene. Genotypes should be coded as 0,1 or 2.
y	a numeric vector with two levels. Phenotype values are coded as 0 or 1.
collapse	a logical value indicating whether or not to collapse all the variants for association testing. Default = FALSE for single variant tests, otherwise TRUE represents collapsed association test.
test	a character to choose the rare variant method that combines with the Zoom-Focus Algorithm.If test = "SKAT", the SKAT of variance component test is applied. If test = "SKATO", the SKAT-O of combination method test is applied. If test = "burden", the weighted burden test is applied. If test = "wtest", the W-test of burden test category is applied.

Value

The output depends on collapse setting:

When collapse=FALSE, it returns a data frame of following components:

pos	Array position/name of variant
odds.ratio	Odds ratio of variant
p.value	P-value of single variant association test

When collapse=TRUE, it returns a table of following elements:

p.value	P-value of gene-based rare variant test
odds.ratio	Regional odds ratio of collapsed variants within genomic region, see odds.ratio for more details
n.variant	Total number of variants within test region
test	Type of selected rare variant method

References

M. H. Wang., R. Sun., et al. (2016) A fast and powerful W-test for pairwise epistasis testing. Nucleic Acids Research.doi:10.1093/nar/gkw347.

R. Sun., H. Weng., et al. (2016) A W-test collapsing method for rare-variant association testing in exome sequencing data. Genetic Epidemiology, 40(7): 591-596.

Wu, M. C., Lee, S., et al. (2011) Rare Variant Association Testing for Sequencing Data Using the Sequence Kernel Association Test (SKAT). American Journal of Human Genetics, 89, 82-93.

Lee, S., Emond, M.J., et al. (2012) Optimal unified approach for rare variant association testing with application to small sample case-control whole-exome sequencing studies. American Journal of Human Genetics, 91, 224-237.

Lee, S., Fuchsberger, C., et al. (2015) An efficient resampling method for calibrating single and gene-based rare variant association analysis in case-control studies. Biostatistics, kxv033.

See Also

fuzzy.zfa, odds.ratio, zfa.plot

Examples

```
data(fuzzy.zfa.example)
attach(fuzzy.zfa.example)

# Single variant tests by wtest
result.single.variants<-test.variants(Z,y,collapse=FALSE,test = "wtest")

# Collapsed association test by SKAT
result.collapsed.variants<-test.variants(Z,y,collapse=TRUE,test = "SKAT")
```

`zfa` *Zoom-Focus Algorithm*

Description

This function performs the Zoom-Focus Algorithm (ZFA) to locate optimal testing regions for rare variant association tests and performs the test based on the optimized regions. The package is suitable to be applied on sequencing data set that is composed of variants with minor allele frequency less than 0.01 (rare variants). The package calls existing rare variant test functions to conduct rare variant test.

ZFA consists of two steps: Zooming and Focusing. In the first step Zooming, a given genomic region is partitioned by an order of two, and the best partition is located using multiple testing corrected p-values returned by desired rare variant test. In the second step Focusing, the boundaries of the zoomed region are refined by allowing them to expand or shrink at micro-level. The computation complexity is linear to the number of variants for Zooming (when the option fast.path=FALSE); a fast-Zoom version can further reduce the complexity to the logarithm of data size (fast.path=TRUE, default).

Usage

```
zfa(data, y, bin = 256, fast.path = TRUE, filter.pval = 0.01,
  output.pval = 0.05, sort = TRUE, CommonRare_Cutoff = 0.01,
  test = c("SKAT", "SKATO", "burden", "wtest"))
```

Arguments

`data`	a data frame or numeric matrix. Genotypes should be coded as 0,1 or 2.
`y`	a numeric vector with two levels. Phenotype values are coded as 0 or 1.
`bin`	a numeric integer taking value of power of two, namely, 2, 4, 8, 16, 32, 64, 128, 256, 512 etc. The bin size specifies the initial window size P to perform the Zoom-Focus Algorithm. Default bin =256.
`fast.path`	a logical value indicating whether or not to use the fast-Zoom approach. The fast-Zoom performs a binary search instead of exhaustive search, such that at each partition order, the region is divided into two parts, only the part with smaller p-value is continued for the next level search. Default = TRUE.
`filter.pval`	a p-value threshold to select zoomed regions for conducting focusing step. When specified, only zoomed regions with p-value smaller than the threshold will be passed to focusing step. Default=0.01. Set filter.pval=NULL for conducting focusing step with all the zoomed regions.

`output.pval` a p-value threshold for filtering the output. If set NULL, all the results will be listed; otherwise, the function will only output the regions with p-values smaller than output.pval. Default=0.05.

`sort` a logical value indicating whether or not to sort the output by p-values in ascending order. Default = TRUE.

`CommonRare_Cutoff` MAF cutoff to define common and rare variants. Default=0.01.

`test` a character to choose the rare variant method that combines with the Zoom-Focus Algorithm.If test = "SKAT", the SKAT of variance component test is applied. If test = "SKATO", the SKAT-O of combination method test is applied. If test = "burden", the weighted burden test is applied. If test = "wtest", the W-test of burden test category is applied.

Details

The algorithm divides sequencing data into multiple fixed genomic regions with a certain initial bin size. ZFA is conducted in each bin. Current version includes 4 existing rare variant tests. The SKAT, SKAT-O and weighted burden test are called from `'SKAT'` package, and the W-test Collapsing method is self-contained.

Value

The `"zfa"` function returns a list with the following components:

`n.regions` Total number of regions to which the input genotype data is divided by initial bin size P.

`n.rare` Total number of rare variants used for the analysis.

`n.common` Total number of common variants excluded in the analysis.

`results` The testing results consist of several elements: 1) `lower.bound` and `upper.bound` represent variant information which indicates the lower and upper bound of optimized testing region; 2) `opt.region.size` denotes the size of optimal testing region after performing ZFA; 3) `corrected.pvalue` displays the multiple testing (Bonferroni) corrected p-value of the optimal testing region.

`Bon.sig.level` Suggested Bonferroni corrected significance level for the input data at threshold alpha=0.05, which equals to 0.05 / # of regions.

`variants` The variants contained in each output optimal region.

Note that the `variants` in the optimized region will not be printed in default. User can can extract the information by calling details of results. See an example in **Examples** section.

The zfa optimizes the testing region according to input variant sequence and assumes that they are arranged by chromosome positions. If variants from non-adjacent genomic regions are input as one data, the zfa will still treat them as adjacent. In this case, the user should be careful in interpreting the results: when an optimized region consists of distant variants, the region may not be biologically meaningful; when an optimized region consists of variants from two neighboring genes, the results may be meaningful.

Author(s)

Haoyi Weng & Maggie Wang

References

M. H. Wang., H. Weng., et al. (2017) A Zoom-Focus algorithm (ZFA) to locate the optimal testing region for rare variant association tests. Bioinformatics.

M. H. Wang., R. Sun., et al. (2016) A fast and powerful W-test for pairwise epistasis testing. Nucleic Acids Research.doi:10.1093/nar/gkw347.

R. Sun., H. Weng., et al. (2016) A W-test collapsing method for rare-variant association testing in exome sequencing data. Genetic Epidemiology, 40(7): 591-596.

Wu, M. C., Lee, S., et al. (2011) Rare Variant Association Testing for Sequencing Data Using the Sequence Kernel Association Test (SKAT). American Journal of Human Genetics, 89, 82-93.

Lee, S., Emond, M.J., et al. (2012) Optimal unified approach for rare variant association testing with application to small sample case-control whole-exome sequencing studies. American Journal of Human Genetics, 91, 224-237.

Lee, S., Fuchsberger, C., et al. (2015) An efficient resampling method for calibrating single and gene-based rare variant association analysis in case-control studies. Biostatistics, kxv033.

Examples

```
data(zfa.example)
attach(zfa.example)

# fast-zoom with wtest, all zoomed regions passed to focusing step, and output all results
zfa.result1<-zfa(X,y,bin = 32,fast.path = TRUE,filter.pval=NULL,output.pval=NULL,test = "wtest")

# zooming with wtest, select zoomed regions for focusing and output regions with both p-value<0.01
zfa.result2<-zfa(X,y,bin = 32,fast.path = FALSE,filter.pval=0.01,output.pval=0.01,test = "wtest")

## an example to view the detail of variants in each output optimal region
result1.detail<-zfa.result1$variants
```

`zfa.example` *Example data for ZFA*

Description

Example data for "`zfa`" function.

Usage

```
data(zfa.example)
```

Format

The `zfa.example` contains the following objects:

`X`

a numeric genotype matrix of 1800 individuals and 512 rare variants. Each row represents a different individual, and each column represents a different rare genetic variant. Genotypes are coded as 0,1 or 2.

`y`

a numeric vector of binary phenotypes, among which there are 343 cases and 1457 controls.

See Also

zfa

`zfa.plot` *OR plot for graphical visualization of fuzzy ZFA optimized region*

Description

This function visualizes the results of optimized region by OR plot in real data analysis. It returns standard ggplot object, upon which the user can add other geoms.

Usage

```
zfa.plot(singleVariantResult, zfaResult, geneName = NULL, bpInfo = NULL,
  chr = NULL, ...)
```

Arguments

`singleVariantResult`	results of single variant test, a data frame with three variables: pos, pvalue and OR. The results could be directly obtained from `"test.variants"` function.
`zfaResult`	results of fuzzy ZFA optimization, the output of `"fuzzy.zfa"` function.
`geneName`	Gene name or symbol of the region, if any. Default = NULL.
`bpInfo`	a vector of bp information for each variant, if any. Default = NULL.
`chr`	a number or character indicating which chromosome the region belongs to, if any. Default = NULL.
`...`	graphical parameters.

Details

The OR plot is designed to gain more biological insights by graphical visualization for the optimized region after applying fuzzy ZFA.

The x axis represents the array and corresponding chromosome position, and the y axis denotes the log transform of odds ratio. Red lines indicate variants of increased risk of disease, while green lines suggest variants with protective effect.

The pink bubbles define the p-value of each rare variant, a smaller p-value gives a larger scale. The optimized region is highlighted with labels of both regional odds ratio and p-value.

See Also

`fuzzy.zfa`, `test.variants`, `odds.ratio`

Examples

```
data(fuzzy.zfa.example)
attach(fuzzy.zfa.example)

# Generate single variant results by wtest
result1<-test.variants(Z,y,collapse=FALSE,test = "wtest")

# Generate zfa optimization results by wtest
result2<-fuzzy.zfa(Z,y,fast.path = TRUE,filter.pval=0.01,test = "wtest")

# Visualize the optimization results by OR plot
OR.plot<-zfa.plot(singleVariantResult=result1,zfaResult=result2)
plot(OR.plot)
```

Index