Package ‘HAP.ROR’

February 19, 2015

Type Package
Title Recursive Organizer (ROR)
Version 1.0
Date 2013-03-23
Author Lue Ping Zhao and Xin Huang
Maintainer Xin Huang <xhuang.fhcrc@gmail.com>
Depends R (>= 2.15.2), hash, ape
Description Functions to perform ROR for sequence-based association analysis
License GPL-2
NeedsCompilation no
Repository CRAN
Date/Publication 2013-03-25 06:57:39

R topics documented:

HAP.ROR-package .................................................. 2
AIC .................................................. 3
case.sub .................................................. 4
cc.sim .................................................. 4
collapse .................................................. 5
ctl.sub .................................................. 6
deletion .................................................. 6
grp.list .................................................. 7
HAPror .................................................. 8
lib.sub .................................................. 9
lib.sub.names ............................................. 12
ODS.ror .................................................. 13
proteinf .................................................. 14

Index 15
HAP.ROR-package

Recursive Organizer (ROR)

Description

functions to perform ROR for sequence-based association analysis

Details

<table>
<thead>
<tr>
<th>Package</th>
<th>HAP.ROR</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type</td>
<td>Package</td>
</tr>
<tr>
<td>Version</td>
<td>1.0</td>
</tr>
<tr>
<td>Date</td>
<td>2013-03-23</td>
</tr>
<tr>
<td>License</td>
<td>GPL-2</td>
</tr>
</tbody>
</table>

Author(s)

Lue Ping Zhao and Xin Huang
Maintainer: Xin Huang <xhuang.fhrc@gmail.com>

References

Zhao, L.P. and Huang, X. Recursive organizer (ROR): an analytic framework for sequence-based association analysis. Human Genetics, 2013

Examples

```r
library("HAP.ROR")
data(case.sub)
data(ctl.sub)
data(lib.sub)
data(lib.sub.names)
ror.res <- HAP.ror(case.sub, ctl.sub, lib.sub, lib.sub.names, alpha=0.01, ref.level="101");
# grouping result:
round(ror.res$dev.list, 2);
round(ror.res$AIC.list, 2);
ror.res$df.list;
ror.res$deleted.snps;
ror.res$grp.result;
ror.res$significant;
# model summary:
ror.res$model.summary;
# output tables and figures used for ror result
data(proteinf)
```
AIC/Deviance calculation

Description

function for AIC/Deviance calculation given index of deleted SNPs

Usage

AIC(case.sub, ctl.sub, lib.sub, lib.sub.names, deleted.snps, ref = "NA")

Arguments

case.sub case subjects, two columns for two haplotypes
ctl.sub control subjects, two columns for two haplotypes
lib.sub the alleles library contains allele sequences for those only appear in the case and control samples
lib.sub.names the corresponding names of the alleles
deleted.snps a vector of positions of deleted SNPs
ref allele names for the reference level, the default reference level (ref="NA") is the most common alleles

Value

logLk log-likelihood
AIC AIC
res the result object return from GLM
dev deviance
df degree of freedom
dev.null deviance for null model
df.null degree of freedom for null model

Author(s)

Xin Huang
**case.sub**  
---  
**Description**  

case samples DRB1 alleles  

**Usage**  

data(case.sub)  

**Format**  

A data frame with 45 observations on the following 2 variables.  
drb1_4digit_1 a numeric vector  
drb1_4digit_2 a numeric vector  

**Examples**  

data(case.sub)

---

**cc.sim**  
---  
**Description**  

simulating case-control data given causal amino acids/haplotype alleles  

**Usage**  

cc.sim(n.ctrl, n.case, beta0, beta1, case.sub, ctl.sub, lib.sub, lib.sub.names, risk.type = "AA", risk.inx = 1, risk.names = c("BSP1", "BSP2"), min.count = 1, pred = "case", pred.type = c("alt", "nom"), pred.inx = 1, lib.name = "lib", lib.type = c("alt", "nom"), lib.inx = 1, lib.names = c("BSP1", "BSP2"), sim = TRUE)  

**Arguments**  

- **n.ctrl**: number of control samples desired to generate  
- **n.case**: number of case samples desired to generate  
- **beta0**: the coefficient of intercept for logistic model  
- **beta1**: the coefficient of the causal SNP for logistic model  
- **case.sub**: case subjects, two columns for two haplotypes  
- **ctl.sub**: control subjects, two columns for two haplotypes  
- **lib.sub**: the alleles library contains allele sequences for those only appear in the case and control samples
**lib.sub.names**  the corresponding names of the alleles

**risk.type**  risk.type="AA": simulated from given amino acid position as shown in matrix
lib.sub, use risk.inx to input position risk.type="allele": simulated from given
risk alleles, use risk.names=c("301", "302") to specified those alleles

**risk.inx**  the given amino acid position

**risk.names**  allele names

**min.count**  use to calculate the warning if the selected alleles have too small frequencies

**ctl.only**  use control only to simulate or not

**Value**

- **y**  phenotype
- **x**  simulated samples
- **risk.names**  the input risk allele names
- **select.freq**  simulated allele frequencies

**Author(s)**

Xin Huang

---

**Description**

function for assign group info to samples after collapsing

**Usage**

collapse(case, ctl, lib, names, snp.de = -1)

**Arguments**

- **case**  case samples: 1st_col=haplotype_1, 2nd_col=haplotype_2
- **ctl**  control samples: 1st_col=haplotype_1, 2nd_col=haplotype_2
- **lib**  the tag-SNPs library *.4d with the only alleles appear in sample
- **names**  corresponding allele names in the same format as appear in sample
- **snp.de**  the column position of a list of SNPs to be deleted, default no delete

**Author(s)**

Xin Huang
ctl.sub  

**control samples of DRB1 alleles**

---

**Description**

control samples of DRB1 alleles

**Usage**

data(ctl.sub)

**Format**

A data frame with 32 observations on the following 2 variables.

- `drb1_4digit_1` a numeric vector
- `drb1_4digit_2` a numeric vector

**Examples**

data(ctl.sub)

---

**deletion**  

**deletion searching**

---

**Description**

function of searching for the next grouping given deleted SNPs

**Usage**

deletion(lib, lib.names, case.sub, ctl.sub, aic.now, dev.now, df.now, rank = FALSE, cut = -1, deleteNsnp = -1, ref = "BnaBL", alpha = "PNPUL", step = PI)

**Arguments**

- `lib` the alleles library contains allele sequences for those only appear in the case and control samples
- `lib.names` the corresponding names of the alleles
- `case.sub` case subjects, two columns for two haplotypes
- `ctl.sub` control subjects, two columns for two haplotypes
- `aic.now` aic for the current model
- `dev.now` deviance for the current model
- `df.now` degree of freedom for the current model
```
grp.list 7

rank  numbers of pairs with top similarity scores to be investigate if FALSE, then
deviance is calculated for the step-wise merger, then option "alpha" and "step"
is used

cut  cutoff for similarity score to be consider, default is -1, means all scores above 0
delete.snp  a vector of position of deleted SNPs
ref  allele names for the reference level, the default reference level (ref="NA") is the
most common alleles
alpha  family-wise error, used for deviance only
step  index for how many deletions have been carried so far

Value

   del  position of deleted SNPs
   aic  AIC
   df  degree of freedom
   dev  deviance
   stop  1: no merge found
   record  the record of the searching path

Author(s)

Xin Huang

Description

  grouping of alleles given deleted SNPs

Usage

  grp.list(allele, snp.de = -1)

Arguments

  allele  data.frame of all the alleles
  snp.de  a vector SNP position to delete

Author(s)

Xin Huang
```
Description

perform ROR for sequence-based association analysis

Usage

\texttt{HAP.ror}(\texttt{case.sub}, \texttt{ctl.sub}, \texttt{lib.sub}, \texttt{lib.sub.names}, \texttt{alpha = 0.01}, \texttt{ref.level = NA}, \texttt{display.proc = TRUE})

Arguments

- \texttt{case.sub}: case subjects, two columns for two haplotypes
- \texttt{ctl.sub}: control subjects, two columns for two haplotypes
- \texttt{lib.sub}: the alleles library contains allele sequences for those only appear in the case and control samples
- \texttt{lib.sub.names}: the corresponding names of the alleles (mapping of full name in the library and short name in samples)
- \texttt{alpha}: significance level
- \texttt{ref.level}: name of the reference allele, "NA" use the most common allele as reference, can also specify allele name, for DRB1, it is "101"
- \texttt{display.proc}: display the grouping process or not? default is TRUE

Details

This function performs ROR for sequence-based association analysis

Value

- \texttt{dev.list}: deviances for all steps of ROR
- \texttt{AIC.list}: AICs for all steps of ROR
- \texttt{df.list}: degree of freedom for all steps of ROR
- \texttt{records}: the record of the whole ROR process
- \texttt{deleted.snps.1s}: the history of SNP deletions for all steps of ROR
- \texttt{deleted.snps}: the final vector of deleted SNPs
- \texttt{grp.result}: the final grouping result
- \texttt{model.summary}: the GLM model summary for the final grouping

Author(s)

Lue Ping Zhao and Xin Huang
Maintainer: Xin Huang <xhuang.fhcrc@gmail.com>
lib.sub

References
Zhao, L.P. and Huang, X. Recursive organizer (ROR): an analytic framework for sequence-based association analysis. Human Genetics, 2013

Examples

library("HAP.ROR")
data(case.sub)
data(ctl.sub)
data(lib.sub)
data(lib.sub.names)
ror.res <- HAP.ror(case.sub, ctl.sub, lib.sub, lib.sub.names, alpha=0.01, ref.level="101");

# grouping result:
round(ror.res$dev.list, 2);
round(ror.res$AIC.list, 2);
ror.res$df.list;
ror.res$deleted.snps;
ror.res$grp.result;
ror.res$significant;
# model summary:
ror.res$model.summary;

---

lib.sub

**DRB1 cDNA sequences**

Description
DRB1 cDNA sequences, with 0 denote the same as reference (DRB1*0101)

Usage
data(lib.sub)

Format
A data frame with 10 DRB1 alleles (those unique alleles in cases + controls) on the following 92 nucleic acid bases. Column names denote the amino acid position. e.g., X.25.1 means the first nucleic acid base on the -25th amino acid, X9.2 means the second nucleic acid base on the 9th amino acid of DRB1 allele, etc

X.25.1 a factor with levels 0 G
X.24 a factor with levels 0 T
X.17 a factor with levels 0 G
X.16.1 a factor with levels 0 T
X.16.2 a factor with levels 0 T
X. 1 a factor with levels Ø T
X4 a factor with levels Ø A
X4. 1 a factor with levels Ø A
X8 a factor with levels Ø C
X9 a factor with levels Ø A C G
X9. 1 a factor with levels Ø A
X9. 2 a factor with levels Ø A
X10. 2 a factor with levels Ø C
X11 a factor with levels Ø A G T
X11. 1 a factor with levels Ø A C G
X12. 2 a factor with levels Ø 3 A C
X13 a factor with levels Ø A C G
X13. 1 a factor with levels Ø A C G
X13. 2 a factor with levels Ø G
X14 a factor with levels Ø A T
X14. 2 a factor with levels Ø G
X16 a factor with levels Ø A T
X18. 2 a factor with levels Ø T
X19. 2 a factor with levels Ø C
X21. 2 a factor with levels Ø A C
X23 a factor with levels Ø 3 4 A G
X23. 2 a factor with levels Ø T
X24. 2 a factor with levels Ø A
X26. 1 a factor with levels Ø A C G
X26. 2 a factor with levels Ø A C T
X28 a factor with levels Ø C
X28. 2 a factor with levels Ø C G
X30 a factor with levels Ø C G
X30. 1 a factor with levels Ø A C T
X30. 2 a factor with levels Ø T
X31 a factor with levels Ø G T
X32 a factor with levels Ø C G
X32. 2 a factor with levels Ø C
X33 a factor with levels Ø C
X34. 2 a factor with levels Ø G
X35. 2 a factor with levels Ø A
X37 a factor with levels Ø A C G
lib.sub

X37. 1 a factor with levels 0 2 A T
X37. 2 a factor with levels 0 0 G
X43. 2 a factor with levels 0 0 T
X44. 2 a factor with levels 0 A C
X47. 1 a factor with levels 0 2 T
X47. 2 a factor with levels 0 0 T
X48. 2 a factor with levels 0 A C
X51. 2 a factor with levels 0 A C T
X52. 2 a factor with levels 0 0 A
X55. 2 a factor with levels 0 0 C
X57. a factor with levels 0 A
X57. 1 a factor with levels 0 0 C G T
X57. 2 a factor with levels 0 0 A C G
X58. 2 a factor with levels 0 0 G T
X59. 2 a factor with levels 0 0 A C
X60. 1 a factor with levels 0 2 0 C
X67. a factor with levels 0 0 2 A G T
X67. 2 a factor with levels 0 0 T
X68. 2 a factor with levels 0 0 A
X69. 2 a factor with levels 0 0 A
X70. 2 a factor with levels 0 0 G
X70. 3 a factor with levels 0 2 0 G
X70. 4 a factor with levels 0 0 A C
X71. a factor with levels 0 0 G T
X71. 1 a factor with levels 0 0 2 A C
X72. 2 a factor with levels 0 0 C T
X73. 1 a factor with levels 0 0 G T
X73. 2 a factor with levels 0 0 A T
X74. a factor with levels 0 0 A C
X74. 1 a factor with levels 0 2 A G T
X75. 2 a factor with levels 0 0 A
X78. a factor with levels 0 0 C G
X78. 1 a factor with levels 0 0 T
X78. 2 a factor with levels 0 A G T
X83. 2 a factor with levels 0 0 T
X86. 1 a factor with levels 0 A C T
X86. 2 a factor with levels 0 0 G
lib.sub.names

X90.2 a factor with levels 0 G
X93  a factor with levels 0 A
X95.2 a factor with levels 0 C
X96  a factor with levels 0 C T
X96.2 a factor with levels 0 A T
X98  a factor with levels 0 G
X120.1 a factor with levels 0 A
X140  a factor with levels 0 A
X179.2 a factor with levels 0 C
X181.2 a factor with levels 0 A
X206.2 a factor with levels 0 T
X217  a factor with levels 0 T
X233.1 a factor with levels 0 G

Examples

data(lib.sub)
## maybe str(lib.sub) ; plot(lib.sub) ...

---

lib.sub.names   unique DRB1 allele names in the sample

Description

unique DRB1 allele names in the sample

Usage

data(lib.sub.names)

Format

The format is: chr [1:10, 1:2] "DRB1*01:01" "DRB1*03:01" "DRB1*04:01" "DRB1*04:04" "DRB1*07:01" "DRB1*08:01" ...

Examples

data(lib.sub.names)
Description

function for output tables and figures related to ROR result

Usage

ODS.ror(case.sub, ctl.sub, lib.sub, lib.sub.names, records, dev.list, AIC.list, deleted.snps.ls, proteinf, loc = Bdrb1, ref.level = B1P1B1)

Arguments

case.sub case subjects, two columns for two haplotypes
ctl.sub control subjects, two columns for two haplotypes
lib.sub the alleles library contains allele sequences for those only appear in the case and control samples
lib.sub.names the corresponding names of the alleles (mapping of full name in the library and short name in samples)
records the record of the whole ROR process
dev.list deviances for all steps of ROR
AIC.list AICs for all steps of ROR
deleted.snps.ls the history of SNP deletions for all steps of ROR
proteinf amino acid matrix for the corresponding alleles
locus name of locus
ref.level name of reference allele

Author(s)

Xin Huang

References

Zhao, L.P. and Huang, X. Recursive organizer (ROR): an analytic framework for sequence-based association analysis. Human Genetics, 2013

Examples

library("HAP.ROR")
data(case.sub)
data(ctl.sub)
data(lib.sub)
data(lib.sub.names)
ror.res <- HAP.ror(case.sub, ctl.sub, lib.sub, lib.sub.names, alpha=0.01, ref.level="101");
# grouping result:
round(ror.res$dev.list, 2);
round(ror.res$AIC.list, 2);
ror.res$df.list;
ror.res$deleted.snps;
ror.res$grp.result;
ror.res$significant;
# model summary:
ror.res$model.summary;
# output tables and figures used for ror result
data(proteinf)
OOD.ror(case.sub=case.sub, ctl.sub=ctl.sub, lib.sub=lib.sub, lib.sub.names=lib.sub.names, records=ror.res$records, aicNlist=ror.resDaicNlistL deletedNsnpsNls=ror.resDdeletedNsnpsNlsL proteinf=proteinfL locus=Bdrb1*BL refNlevel=B1P1BI);
cat("ROR result tables/figures are store in folder:", getwd(),"\n")

---

<table>
<thead>
<tr>
<th>proteinf</th>
<th>DRB1 amino acid sequences</th>
</tr>
</thead>
</table>

**Description**

DRB1 amino acid sequences, with 0 denote the same as reference (DRB1*0101)

**Usage**

data(proteinf)

**Format**

A data frame with 1052 observations on the following 269 variables. Column names denote the amino acid position. e.g., X.25.1 means the first nucleic acid base on the -25th amino acid, X9.2 means the second nucleic acid base on the 9th amino acid of DRB1 allele, etc

**Examples**

data(proteinf)
## maybe str(proteinf) ; plot(proteinf) ...
Index

*Topic \textasciitilde kwd1
AIC, 3
cc.sim, 4
collapse, 5
deletion, 6
grp.list, 7
HAP.ro, 8
ODS.ror, 13

*Topic \textasciitilde kwd2
AIC, 3
cc.sim, 4
collapse, 5
deletion, 6
grp.list, 7
HAP.ro, 8
ODS.ror, 13

*Topic datasets
case.sub, 4
ctl.sub, 6
lib.sub, 9
lib.sub.names, 12
proteinf, 14

AIC, 3
case.sub, 4
cc.sim, 4
collapse, 5
ctl.sub, 6
deletion, 6
grp.list, 7
HAP.ROR (HAP.ROR-package), 2
HAP.ro, 8
HAP.ROR-package, 2
lib.sub, 9
lib.sub.names, 12