

Package ‘samr’

April 19, 2009

Title SAM: Significance Analysis of Microarrays

Version 1.26

Author R. Tibshirani, G. Chu, T. Hastie, Balasubramanian Narasimhan

Description Significance Analysis of Microarrays

Maintainer Rob Tibshirani <tibs@stat.stanford.edu>

Depends impute

License LGPL

URL <http://www-stat.stanford.edu/~tibs/SAM>

Repository CRAN

Date/Publication 2008-09-28 18:30:46

R topics documented:

samr	2
samr.assess.samplesize	8
samr.assess.samplesize.plot	10
samr.compute.delta.table	11
samr.compute.siggenes.table	12
samr.missrate	14
samr.plot	15
samr.pvalues.from.perms	16
samr.tail.strength	17

Index	19
--------------	-----------

samr

*Significance analysis of microarrays***Description**

Correlates a large number of features (eg genes) with an outcome variable, such as a group indicator, quantitative variable or survival time

Usage

```

samr(data, resp.type=c("Quantitative", "Two class unpaired", "Survival", "Multiclass",
"One class timecourse", "Two class paired timecourse", "Pattern discovery"), s0=NULL

```

Arguments

<code>data</code>	Data object with components x- p by n matrix of features, one observation per column (missing values allowed); y- n-vector of outcome measurements; censoring.status- n-vector of censoring censoring.status (1= died or event occurred, 0=survived, or event was censored), needed for a censored survival outcome
<code>resp.type</code>	Problem type: "quantitative" for a continuous parameter; "Two class unpaired" ; "Survival" for censored survival outcome; "Multiclass" : more than 2 groups; "One class" for a single group; "Two class paired" for two classes with paired observations; "Two class unpaired timecourse", "One class time course", "Two class.paired timecourse" or "Pattern discovery"
<code>s0</code>	Exchangeability factor for denominator of test statistic; Default is automatic choice
<code>s0.perc</code>	Percentile of standard deviation values to use for s0; default is automatic choice; -1 means s0=0 (different from s0.perc=0, meaning s0=zeroeth percentile of standard deviation values= min of sd values
<code>nperms</code>	Number of permutations used to estimate false discovery rates
<code>center.arrays</code>	Should the data for each sample (array) be median centered at the outset? Default =FALSE
<code>testStatistic</code>	Test statistic to use in two class unpaired case.Either "standard" (t-statistic) or ,"wilcoxon" (Two-sample wilcoxon or Mann-Whitney test)
<code>time.summary.type</code>	Summary measure for each time course: "slope", or "signed.area"),
<code>regression.method</code>	Regression method for quantitative case: "standard", (linear least squares) or "ranks" (linear least squares on ranked data)
<code>return.x</code>	Should the matrix of feature values be returned? Only useful for time course data, where x contains summaries of the features over time. Otherwise x is the same as the input data data\$x

<code>knn.neighbors</code>	Number of nearest neighbors to use for imputation of missing features values
<code>random.seed</code>	Optional initial seed for random number generator (integer)
<code>xl.mode</code>	Used by Excel interface
<code>xl.time</code>	Used by Excel interface
<code>xl.prevfit</code>	Used by Excel interface

Details

Carries out a SAM analysis. Applicable to microarray data and other data with a large number of features. This is the R package that is called by the "official" SAM Excel package v2.0. The format of the response vector y and the calling sequence is illustrated in the examples below. A more complete description is given in the SAM manual at <http://www-stat.stanford.edu/~tibs/SAM>

Value

A list with components

<code>n</code>	Number of observations
<code>x</code>	Data matrix p by n (p =# genes or features). Equal to the matrix <code>data\$x</code> in the original call to <code>samr</code> except for (1) time course analysis, where it contains the summarized data or (2) quantitative outcome with rank regression, where it contains the data transformed to ranks. Hence it is null except for in time course analysis.
<code>y</code>	Vector of n outcome values. equal the values <code>data\$y</code> in the original call to <code>samr</code> , except for (1) time course analysis, where it contains the summarized y or (2) quantitative outcome with rank regression, where it contains the y values transformed to ranks
<code>argy</code>	The values <code>data\$y</code> in the original call to <code>samr</code>
<code>censoring.status</code>	Censoring status indicators if applicable
<code>testStatistic</code>	Test Statistic used
<code>nperms</code>	Number of permutations requested
<code>nperms.act</code>	Number of permutations actually used. Will be $< nperms$ when # of possible permutations $\leq nperms$ (in which case all permutations are done)
<code>tt</code>	$tt = \text{numer}/sd$, the vector of p test statistics for original data
<code>numer</code>	Numerators for <code>tt</code>
<code>sd</code>	Denominators for <code>tt</code> . Equal to standard deviation for feature plus <code>s0</code>
<code>s0</code>	Computed exchangeability factor
<code>s0.perc</code>	Computed percentile of standard deviation values. $s0 = s0.perc$ percentile of the gene standard deviations
<code>eva</code>	p -vector of expected values for <code>tt</code> under permutation sampling
<code>perms</code>	<code>nperms.act</code> by n matrix of permutations used. Each row is a permutation of $1, 2, \dots, n$

<code>permsy</code>	<code>nperms.act</code> by <code>n</code> matrix of permutations used. Each row is a permutation of <code>y1,y2,...yn</code> . Only one of <code>perms</code> or <code>permsy</code> is non-Null, depending on <code>resp.type</code>
<code>all.perms.flag</code>	Were all possible permutations used?
<code>ttstar</code>	<code>p</code> by <code>nperms.aca</code> matrix <code>t</code> of test statistics from permuted data. Each column if sorted in descending order
<code>ttstar0</code>	<code>p</code> by <code>nperms.act</code> matrix of test statistics from permuted data. Columns are in order of data
<code>eigengene.number</code>	The number of the eigengene (eg 1,2,..) that was requested for Pattern discovery
<code>eigengene</code>	Computed eigengene
<code>pi0</code>	Estimated proportion of non-null features (genes)
<code>foldchange</code>	<code>p</code> -vector of foldchanges for original data
<code>foldchange.star</code>	<code>p</code> by <code>nperms.act</code> matrix estimated foldchanges from permuted data
<code>sdstar.keep</code>	<code>n</code> by <code>nperms.act</code> matrix of standard deviations from each permutation
<code>censoring.status.star.keep</code>	<code>n</code> by <code>nperms.act</code> matrix of <code>censoring.status</code> indicators from each permutation
<code>resp.type</code>	The response type used. Same as <code>resp.type.arg</code> , except for time course data, where time data is summarized and then treated as non-time course. Eg if <code>resp.type.arg="oneclass.timecourse"</code> then <code>resp.type="oneclass"</code>
<code>resp.type.arg</code>	The response type requested in the call to <code>samr</code>
<code>stand.contrasts</code>	For multiclass data, <code>p</code> by <code>nclass</code> matrix of standardized differences between the class mean and the overall mean
<code>stand.contrasts.star</code>	For multiclass data, <code>p</code> by <code>nclass</code> by <code>nperms.act</code> array of standardized contrasts for permuted datasets
<code>stand.contrasts.95</code>	For multiclass data, 2.5 of standardized contrasts. Useful for determining which class contrast for significant genes, are large
<code>call</code>	calling sequence

Author(s)

Balasubramanian Narasimhan and Robert Tibshirani

References

Tusher, V., Tibshirani, R. and Chu, G. (2001): Significance analysis of microarrays applied to the ionizing radiation response PNAS 2001 98: 5116-5121, (Apr 24). <http://www-stat.stanford.edu/~tibs/SAM>

Examples

```
##### two class unpaired comparison
# y must take values 1,2

set.seed(100)
x<-matrix(rnorm(1000*20),ncol=20)
dd<-sample(1:1000,size=100)

u<-matrix(2*rnorm(100),ncol=10,nrow=100)
x[dd,11:20]<-x[dd,11:20]+u
y<-c(rep(1,10),rep(2,10))

data=list(x=x,y=y, geneid=as.character(1:nrow(x)),genenames=paste("g",as.character(1:nrow(x)),s

samr.obj<-samr(data, resp.type="Two class unpaired", nperms=100)

delta=.4
samr.plot(samr.obj,delta)

delta.table <- samr.compute.delta.table(samr.obj)

siggenes.table<-samr.compute.siggenes.table(samr.obj,delta, data, delta.table)

##### two class paired

# y must take values -1, 1, -2,2 etc, with (-k,k) being a pair

set.seed(100)
x<-matrix(rnorm(1000*20),ncol=20)
dd<-sample(1:1000,size=100)

u<-matrix(2*rnorm(100),ncol=10,nrow=100)
x[dd,11:20]<-x[dd,11:20]+u
y=c(-(1:10),1:10)

d=list(x=x,y=y, geneid=as.character(1:nrow(x)),genenames=paste("g",as.character(1:nrow(x)),s

samr.obj<-samr(d, resp.type="Two class paired", nperms=100)

#####quantitative response

# y must take numeric values

set.seed(84048)
x=matrix(rnorm(1000*9),ncol=9)

mu=c(3,2,1,0,0,0,1,2,3)
b=runif(100)+.5
```

```

x[1:100,]=x[1:100,]+ b

y=mu

d=list(x=x,y=y,
geneid=as.character(1:nrow(x)),genenames=paste("gene", as.character(1:nrow(x))))

samr.obj =samr(d, resp.type="Quantitative", nperms=50)

##### oneclass
# y is a vector of ones

set.seed(100)
x<-matrix(rnorm(1000*20),ncol=20)
dd<-sample(1:1000,size=100)

u<-matrix(2*rnorm(100),ncol=10,nrow=100)
x[dd,11:20]<-x[dd,11:20]+u

y<-c(rep(1,20))

data=list(x=x,y=y, geneid=as.character(1:nrow(x)),genenames=paste("g",as.character(1:nrow(x)))

samr.obj<-samr(data, resp.type="One class", nperms=100)

#####survival data
# y is numeric; censoring.status=1 for failures, and 0 for censored

set.seed(84048)
x=matrix(rnorm(1000*50),ncol=50)
x[1:50,26:50]= x[1:50,26:50]+2
x[51:100,26:50]= x[51:100,26:50]-2

y=abs(rnorm(50))
y[26:50]=y[26:50]+2
censoring.status=sample(c(0,1),size=50,replace=TRUE)
d=list(x=x,y=y,censoring.status=censoring.status,
geneid=as.character(1:1000),genenames=paste("gene", as.character(1:1000)))

samr.obj=samr(d, resp.type="Survival", nperms=20)

#####multi-class example
# y takes values 1,2,3,...k where k= number of classes

set.seed(84048)
x=matrix(rnorm(1000*10),ncol=10)
x[1:50,6:10]= x[1:50,6:10]+2
x[51:100,6:10]= x[51:100,6:10]-2

y=c(rep(1,3),rep(2,3),rep(3,4))
d=list(x=x,y=y,geneid=as.character(1:1000),genenames=paste("gene", as.character(1:1000)))

```

```

samr.obj <- samr(d, resp.type="Multiclass")

##### timecourse data

# elements of y are of the form kTimet where k is the class label and t
# is the time; in addition, the suffixes Start or End indicate the first
# and last observation in a given time course
# the class label can be that for a two class unpaired, one class or
# two class paired problem

set.seed(8332)
y=paste(c(rep(1,15),rep(2,15)), "Time", rep(c(1,2,3,4,5,1.1,2.5, 3.7, 4.1,5.5),3), sep="")
start=c(1,6,11,16,21,26)
for(i in start){
y[i]=paste(y[i], "Start", sep="")
}
for(i in start+4){
y[i]=paste(y[i], "End", sep="")
}
x=matrix(rnorm(1000*30), ncol=30)
x[1:50,16:20]=x[1:50,16:20]+matrix(3*c(0,1,2,3,4), ncol=5, nrow=50, byrow=TRUE)
x[1:50,21:25]=x[1:50,21:25]+matrix(3*c(0,1,2,3,4), ncol=5, nrow=50, byrow=TRUE)
x[1:50,26:30]=x[1:50,26:30]+matrix(3*c(0,1,2,3,4), ncol=5, nrow=50, byrow=TRUE)

x[51:100,16:20]=x[51:100,16:20]-matrix(3*c(0,1,2,3,4), ncol=5, nrow=50, byrow=TRUE)
x[51:100,21:25]=x[51:100,21:25]-matrix(3*c(0,1,2,3,4), ncol=5, nrow=50, byrow=TRUE)
x[51:100,26:30]=x[51:100,26:30]-matrix(3*c(0,1,2,3,4), ncol=5, nrow=50, byrow=TRUE)

data=list(x=x,y=y, geneid=as.character(1:nrow(x)), genenames=paste("g", as.character(1:nrow(x)))

samr.obj<- samr(data, resp.type="Two class unpaired timecourse",
  nperms=100, time.summary.type="slope")

##### pattern discovery
# here there is no outcome y; the desired eigengene is indicated by
# the argument eigengene.numbe in the data object

set.seed(32)
x=matrix(rnorm(1000*9), ncol=9)
mu=c(3,2,1,0,0,0,1,2,3)
b=3*runif(100)+.5
x[1:100,]=x[1:100,]+ b

d=list(x=x, eigengene.number=1,
  geneid=as.character(1:nrow(x)), genenames=paste("gene", as.character(1:nrow(x))))

samr.obj=samr(d, resp.type="Pattern discovery", nperms=50)

```

```
samr.assess.samplesize
```

Assess the sample size for a SAM analysis

Description

Estimate the false discovery rate, false negative rate, power and type I error for a SAM analysis. Currently implemented only for two class (unpaired or paired), one-sample and survival problems).

Usage

```
samr.assess.samplesize(samr.obj, data, dif, samplesize.factors=c(1,2,3,5), min.genes
```

Arguments

<code>samr.obj</code>	Object returned from call to <code>samr</code>
<code>data</code>	Data list, same as that passed to <code>samr.train</code>
<code>dif</code>	Change in gene expression between groups 1 and 2, for genes that are differentially expressed. For log base 2 data, a value of 1 means a 2-fold change. For One-sample problems, <code>dif</code> is the number of units away from zero for differentially expressed genes. For survival data, <code>dif</code> is the numerator of the Cox score statistic (this info is provided in the output of <code>samr</code>).
<code>samplesize.factors</code>	Integer vector of length 4, indicating the sample sizes to be examined. The values are factors that multiply the original sample size. So the value 1 means a sample size of <code>ncol(datax)</code> , <code>2meansasamplesizeofncol(datax)</code> , etc.
<code>min.genes</code>	Minimum number of genes that are assumed to truly changed in the population
<code>max.genes</code>	Maximum number of genes that are assumed to truly changed in the population

Details

Estimates false discovery rate, false negative rate, power and type I error for a SAM analysis. The argument `samplesize.factor` allows the use to assess the effect of varying the sample size (total number of samples). A detailed description of this calculation is given in the SAM manual at <http://www-stat.stanford.edu/~tibs/SAM>

Value

A list with components

<code>Results</code>	A matrix with columns: number of genes- both the number differentially expressed genes in the population and number called significant; <code>cutpoint</code> - the threshold used for the absolute SAM score <code>d</code> ; <code>FDR</code> , <code>1-power</code> - the median false discovery rate, also equal to the power for each gene; <code>FDR-90perc</code> - the upper 90th percentile of the FDR; <code>FNR</code> , <code>Type 1 error</code> - the false negative rate, also equal to the type I error for each gene; <code>FNR-90perc</code> - the upper 90th percentile of the FNR
----------------------	---

dif.call	Change in gene expression between groups 1 and 2, that was provided in the call to samr.assess.samplesize
difm	The average difference in SAM score d for the genes differentially expressed vs unexpressed
samplesize.factor	The samplesize.factor that was passed to samr.assess.samplesize
n	Number of samples in input data (i.e. ncol of x component in data)

Author(s)

Balasubramanian Narasimhan and Robert Tibshirani

References

Tusher, V., Tibshirani, R. and Chu, G. (2001): Significance analysis of microarrays applied to the ionizing radiation response" PNAS 2001 98: 5116-5121, (Apr 24). <http://www-stat.stanford.edu/~tibs/sam>

Taylor, J., Tibshirani, R. and Efron. B. (2005). The "Miss rate" for the analysis of gene expression data. Biostatistics 2005 6(1):111-117.

A more complete description is given in the SAM manual at <http://www-stat.stanford.edu/~tibs/SAM>

Examples

```
#generate some example data
set.seed(100)
x<-matrix(rnorm(1000*20),ncol=20)
dd<-sample(1:1000,size=100)

u<-matrix(2*rnorm(100),ncol=10,nrow=100)
x[dd,11:20]<-x[dd,11:20]+u

y<-c(rep(1,10),rep(2,10))

data=list(x=x,y=y, geneid=as.character(1:nrow(x)),genenames=paste("g",as.character(1:nrow(x)))

log2=function(x){log(x)/log(2)}

# run SAM first
samr.obj<-samr(data, resp.type="Two class unpaired", nperms=100)

# assess current sample size (20), assuming 1.5fold difference on log base 2 scale
samr.assess.samplesize.obj<- samr.assess.samplesize(samr.obj, data, log2(1.5))

# assess the effect of doubling the sample size
samr.assess.samplesize.obj2<- samr.assess.samplesize(samr.obj, data, log2(1.5))
```

```
samr.assess.samplesize.plot
```

Make a plot of the results from samr.assess.samplesize

Description

Plots of the results from samr.assess.samplesize

Usage

```
samr.assess.samplesize.plot(samr.assess.samplesize.obj, logx=TRUE, call.win.metafil
```

Arguments

```
samr.assess.samplesize.obj
```

Object returned from call to samr.assess.samplesize

```
logx
```

Should logs be used on the horizontal (# of genes) axis? Default TRUE

```
call.win.metafile
```

Used by Excel interface

Details

Plots results: FDR (or 1-power) and FNR (or 1-type 1 error) from samr.assess.samplesize

Author(s)

Balasubrimanian Narasimhan and Robert Tibshirani

References

Tusher, V., Tibshirani, R. and Chu, G. (2001): Significance analysis of microarrays applied to the ionizing radiation response" PNAS 2001 98: 5116-5121, (Apr 24). <http://www-stat.stanford.edu/~tibs/sam>

Examples

```
#generate some example data
set.seed(100)
x<-matrix(rnorm(1000*20),ncol=20)
dd<-sample(1:1000,size=100)

u<-matrix(2*rnorm(100),ncol=10,nrow=100)
x[dd,11:20]<-x[dd,11:20]+u

y<-c(rep(1,10),rep(2,10))

data=list(x=x,y=y, geneid=as.character(1:nrow(x)),genenames=paste("g",as.character(1:nrow(x)))

log2=function(x){log(x)/log(2)}
```

```
# run SAM first
samr.obj<-samr(data, resp.type="Two class unpaired", nperms=100)

# assess current sample size (20), assuming 1.5fold difference on the log base 2 scale

samr.assess.samplesize.obj<- samr.assess.samplesize(samr.obj, data, log2(1.5))

samr.assess.samplesize.plot(samr.assess.samplesize.obj)
```

```
samr.compute.delta.table
      Compute delta table for SAM analysis
```

Description

Computes tables of thresholds, cutpoints and corresponding False Discovery rates for SAM (Significance analysis of microarrays) analysis

Usage

```
samr.compute.delta.table(samr.obj, min.foldchange=0, dels=NULL, nvals=50)
```

Arguments

<code>samr.obj</code>	Object returned from call to <code>samr</code>
<code>min.foldchange</code>	The minimum fold change desired; should be >1 ; default is zero, meaning no fold change criterion is applied
<code>dels</code>	vector of delat valeus used. By default, 50 values are chosen in the relevant operating chnage for delta. Delta is vertical the distance from the 45 degree line to the upper and lower parallel lines that define the SAM threshold rule.
<code>nvals</code>	Number of delta values used

Details

Returns a table of the FDR and upper and lower cutpoints for various values of delta, for a SAM analysis.

Author(s)

Balasubramanian Narasimhan and Robert Tibshirani

References

Tusher, V., Tibshirani, R. and Chu, G. (2001): Significance analysis of microarrays applied to the ionizing radiation response" PNAS 2001 98: 5116-5121, (Apr 24). <http://www-stat.stanford.edu/~tibs/sam>

Examples

```
#generate some example data
set.seed(100)
x<-matrix(rnorm(1000*20),ncol=20)
dd<-sample(1:1000,size=100)

u<-matrix(2*rnorm(100),ncol=10,nrow=100)
x[dd,11:20]<-x[dd,11:20]+u

y<-c(rep(1,10),rep(2,10))

data=list(x=x,y=y, geneid=as.character(1:nrow(x)),genenames=paste("g",as.character(1:nrow(x)))

samr.obj<-samr(data, resp.type="Two class unpaired", nperms=50)

delta.table<- samr.compute.delta.table(samr.obj)
```

```
samr.compute.siggenes.table
      Compute significant genes table
```

Description

Computes significant genes table, starting with samr object "samr.obj" and delta.table "delta.table"

Usage

```
samr.compute.siggenes.table(samr.obj, del, data, delta.table, min.foldchange=0, al
```

Arguments

samr.obj	Object returned from call to samr
del	Value of delta to define cutoff rule
data	Data object, same as that used in call to samr
delta.table	Object returned from call to samr.compute.delta.table
min.foldchange	The minimum fold change desired; should be >1; default is zero, meaning no fold change criterion is applied
all.genes	Should all genes be listed? Default FALSE
compute.localfdr	Should the local fdrs be computed (this can take some time)? Default FALSE

Value

```
return(list(genes.up=res.up, genes.lo=res.lo, color.ind.for.multi=color.ind.for.multi, ngenes.up=ngenes.up,
ngenes.lo=ngenes.lo))
```

gene.up Matrix of significant genes having positive correlation with the outcome

gene.lo Matrix of significant genes having negative correlation with the outcome

color.ind.for.multi

For multiclass response: a matrix with entries +1 if the class mean is larger than the overall mean at the 95 levels, -1 if less, and zero otherwise. This is useful in determining which class or classes causes a feature to be significant

ngenes.up Number of significant genes with positive correlation

ngenes.lo Number of significant genes with negative correlation

Author(s)

Balasubrimanian Narasimhan and Robert Tibshirani

References

Tusher, V., Tibshirani, R. and Chu, G. (2001): Significance analysis of microarrays applied to the ionizing radiation response" PNAS 2001 98: 5116-5121, (Apr 24). <http://www-stat.stanford.edu/~tibs/sam>

Examples

```
#generate some example data
set.seed(100)
x<-matrix(rnorm(1000*20),ncol=20)
dd<-sample(1:1000,size=100)

u<-matrix(2*rnorm(100),ncol=10,nrow=100)
x[dd,11:20]<-x[dd,11:20]+u

y<-c(rep(1,10),rep(2,10))

data=list(x=x,y=y, geneid=as.character(1:nrow(x)),genenames=paste("g",as.character(1:nrow(x)))

samr.obj<-samr(data, resp.type="Two class unpaired", nperms=100)

delta.table<-samr.compute.delta.table(samr.obj)
del<- 0.3
siggenes.table<- samr.compute.siggenes.table(samr.obj, del, data, delta.table)
```

samr.missrate *Estimate the miss rate table for a SAM analysis*

Description

Estimates the miss rate table, showing the local false negative rate, for a SAM analysis

Usage

```
samr.missrate(samr.obj, del, delta.table, quant=NULL)
```

Arguments

samr.obj	Object returned from call to samr
del	Value of delta to define cutoff rule
delta.table	Object returned from call to samr.compute.delta.table
quant	Optional vector of quantiles to used in the miss rate calculation

Author(s)

Balasubrimanian Narasimhan and Robert Tibshirani

References

Tusher, V., Tibshirani, R. and Chu, G. (2001): Significance analysis of microarrays applied to the ionizing radiation response" PNAS 2001 98: 5116-5121, (Apr 24). <http://www-stat.stanford.edu/~tibs/sam>

Taylor, J., Tibshirani, R. and Efron. B. (2005). The "Miss rate" for the analysis of gene expression data. Biostatistics 2005 6(1):111-117.

Examples

```
#generate some example data
set.seed(100)
x<-matrix(rnorm(1000*20),ncol=20)
dd<-sample(1:1000,size=100)

u<-matrix(2*rnorm(100),ncol=10,nrow=100)
x[dd,11:20]<-x[dd,11:20]+u

y<-c(rep(1,10),rep(2,10))

data=list(x=x,y=y, geneid=as.character(1:nrow(x)),genenames=paste("g",as.character(1:nrow(x)))

samr.obj<-samr(data, resp.type="Two class unpaired", nperms=100)

delta.table<-samr.compute.delta.table(samr.obj)
del<- 0.3
siggenes.table<- samr.compute.siggenes.table(samr.obj, del, data, delta.table)
```

```
samr.missrate(samr.obj, del, delta.table)
```

```
samr.plot
```

Make Q-Q plot for SAM analysis

Description

Makes the Q-Q plot for a SAM analysis

Usage

```
samr.plot(samr.obj, del, min.foldchange=0)
```

Arguments

<code>samr.obj</code>	Object returned from call to <code>samr</code>
<code>del</code>	Value of delta to use. Delta is the vertical distance from the 45 degree line to the upper and lower parallel lines that define the SAM threshold rule.
<code>min.foldchange</code>	The minimum fold change desired; should be >1; default is zero, meaning no fold change criterion is applied

Details

Creates the Q-Q plot from a SAM analysis, marking features (genes) that are significant, i.e. lie outside a slab around the 45 degree line of width delta. A gene must also pass the `min.foldchange` criterion to be called significant, if this criterion is specified in the call.

Author(s)

Balasubramanian Narasimhan and Robert Tibshirani

References

Tusher, V., Tibshirani, R. and Chu, G. (2001): Significance analysis of microarrays applied to the ionizing radiation response" PNAS 2001 98: 5116-5121, (Apr 24). <http://www-stat.stanford.edu/~tibs/sam>

Examples

```
#generate some example data
set.seed(100)
x<-matrix(rnorm(1000*20), ncol=20)
dd<-sample(1:1000, size=100)

u<-matrix(2*rnorm(100), ncol=10, nrow=100)
x[dd, 11:20]<-x[dd, 11:20]+u
```

```

y<-c(rep(1,10),rep(2,10))

data=list(x=x,y=y, geneid=as.character(1:nrow(x)),genenames=paste("g",as.character(1:nrow(x))

samr.obj<-samr(data, resp.type="Two class unpaired", nperms=50)

samr.plot(samr.obj, del=.3)

```

```
samr.pvalues.from.perms
```

Report estimated p-values for each gene, from a SAM analysis

Description

Report estimated p-values for each gene, from set of permutations in a SAM analysis

Usage

```
samr.pvalues.from.perms(tt, ttstar)
```

Arguments

tt	Vector of gene scores, returned by samr in component tt
ttstar	Matrix of gene scores (p by nperm) from nperm permutations. Returned by samr in component ttstar

Author(s)

Balasubrimanian Narasimhan and Robert Tibshirani

References

Taylor, J. and Tibshirani, R. (2005): A tail strength measure for assessing the overall significance in a dataset. Submitted.

Examples

```

#generate some example data
set.seed(100)
x<-matrix(rnorm(1000*20),ncol=20)
dd<-sample(1:1000,size=100)

u<-matrix(2*rnorm(100),ncol=10,nrow=100)
x[dd,11:20]<-x[dd,11:20]+u

y<-c(rep(1,10),rep(2,10))

data=list(x=x,y=y, geneid=as.character(1:nrow(x)),genenames=paste("g",as.character(1:nrow(x))
samr.obj<-samr(data, resp.type="Two class unpaired", nperms=100)

```

```
pv=samr.pvalues.from.perms(samr.obj$tt, samr.obj$ttstar)
```

`samr.tail.strength` *Estimate tail strength for a dataset, from a SAM analysis*

Description

Estimate tail strength for a dataset, from a SAM analysis

Usage

```
samr.tail.strength(samr.obj)
```

Arguments

`samr.obj` Object returned by `samr`

Value

A list with components

`ts` Estimated tail strength. A number less than or equal to 1. Zero means all genes are null; 1 means all genes are differentially expressed.

`se.ts` Estimated standard error of tail strength.

Author(s)

Balasubrimanian Narasimhan and Robert Tibshirani

References

Taylor, J. and Tibshirani, R. (2005): A tail strength measure for assessing the overall significance in a dataset. Submitted.

Examples

```
#generate some example data
set.seed(100)
x<-matrix(rnorm(1000*20), ncol=20)
dd<-sample(1:1000, size=100)

u<-matrix(2*rnorm(100), ncol=10, nrow=100)
x[dd, 11:20]<-x[dd, 11:20]+u

y<-c(rep(1, 10), rep(2, 10))

data=list(x=x, y=y, geneid=as.character(1:nrow(x)), genenames=paste("g", as.character(1:nrow(x)))
```

```
samr.obj<-samr(data, resp.type="Two class unpaired", nperms=100)
samr.tail.strength(samr.obj)
```

Index

*Topic **nonparametric**

- samr, 1
- samr.assess.samplesize, 7
- samr.assess.samplesize.plot, 9
- samr.compute.delta.table, 11
- samr.compute.siggenes.table, 12
- samr.misstrate, 13
- samr.plot, 14
- samr.pvalues.from.perms, 15
- samr.tail.strength, 16

*Topic **survival**

- samr, 1
- samr.assess.samplesize, 7
- samr.assess.samplesize.plot, 9
- samr.compute.delta.table, 11
- samr.compute.siggenes.table, 12
- samr.misstrate, 13
- samr.plot, 14
- samr.pvalues.from.perms, 15
- samr.tail.strength, 16

*Topic **ts**

- samr, 1
- samr.assess.samplesize, 7
- samr.assess.samplesize.plot, 9
- samr.compute.delta.table, 11
- samr.compute.siggenes.table, 12
- samr.misstrate, 13
- samr.plot, 14

*Topic **univar**

- samr, 1
- samr.assess.samplesize, 7
- samr.assess.samplesize.plot, 9

- samr.compute.delta.table, 11
- samr.compute.siggenes.table, 12
- samr.misstrate, 13
- samr.plot, 14
- samr.pvalues.from.perms, 15
- samr.tail.strength, 16

- samr, 1
- samr.assess.samplesize, 7
- samr.assess.samplesize.plot, 9
- samr.compute.delta.table, 11
- samr.compute.siggenes.table, 12
- samr.misstrate, 13
- samr.plot, 14
- samr.pvalues.from.perms, 15
- samr.tail.strength, 16