# Package 'SNAGEE'

April 17, 2025

**Version** 1.48.0

**Date** 2013-07-16

**Title** Signal-to-Noise applied to Gene Expression Experiments

**Author** David Venet <davenet@ulb.ac.be>

**Maintainer** David Venet <davenet@ulb.ac.be>

**Depends** R (>= 2.6.0), SNAGEEdata

**Suggests** ALL, hgu95av2.db

**Enhances** parallel

**Description** Signal-to-Noise applied to Gene Expression Experiments.
Signal-to-noise ratios can be used as a proxy for quality of gene expression studies and samples. The SNRs can be calculated on any gene expression data set as long as gene IDs are available, no access to the raw data files is necessary. This allows to flag problematic studies and samples in any public data set.

**License** Artistic-2.0

**biocViews** Microarray, OneChannel, TwoChannel, QualityControl

**URL** http://bioconductor.org/

**git_url** https://git.bioconductor.org/packages/SNAGEE

**git_branch** RELEASE_3_21

**git_last_commit** ac5a58d

**git_last_commit_date** 2025-04-15

**Repository** Bioconductor 3.21

**Date/Publication** 2025-04-16

## Contents

SNAGEE-package *Signal-to-Noise Applied to Gene Expression Experiments*

## Description

Signal-to-Noise Applied to Gene Expression Experiments

## Details

| | |
|---|---|
| Package: | SNAGEE |
| Version: | 0.99.0 |
| Date: | 2012-01-26 |
| Depends: | R (>= 2.6.0) |
| Imports: | SNAGEEdata |
| Suggests: | ALL |
| Enhances: | parallel |
| License: | Artistic-2.0 |
| URL: | http://fleming.ulb.ac.be/SNAGEE |

Index:



## Author(s)

David Venet <davenet@ulb.ac.be>

Maintainer: David Venet <davenet@ulb.ac.be>

## Examples

```
# Get the list of genes
geneList = getCC()$g;
# Create a random data set
d=list(genes=geneList, data=matrix(rnorm(length(geneList)*50),ncol=50));
# Calculate its quality (it's going to be very close to 0)
qualStudy(d, disattenuate=FALSE);
# Calcuate individual sample qualities
qs = qualSample(d);
```

qualSample *Quality of samples in a study*

## Description

Calculate the relative quality of all samples from a study.

## Usage

```
qualSample(data,mode="complete",cc=NULL,multicore=FALSE)
```

## Arguments

| | |
|---|---|
| data | The study data. If an Eset, toSnageeFormat is called on it. Otherwise, must be a list with fields 'genes' containing the vector of gene IDs (from Entrez) and 'data' containing the gene expression data. |
| mode | Which gene-gene correlation matrix should be used. Can be 'complete' (using all platforms) or 'woAffy' (without the Affy platforms). |
| cc | Can be used if wishing to use a custom gene-gene correlation matrix. Must be a list with fields 'g' containing the gene IDs and 'cc' containing the (upper triangular part of the) correlations. |
| multicore | Should the parallel version be used? This is based on the parallel package, if that package cannot be loaded it will fall back on single core, with a warning. |

## Details

The function calculates the quality of all samples in a study. Lower values are of lower quality. The numerical values of the study (the 'data' field) should be in log-scale, and normalized. It is recommended to used medpolish on the data.

Each gene should only appear once in the gene list. Duplicated genes must be merged before using the function. Non-finite values should also be removed first (using the impute package for instance).

## See Also

SNAGEE, qualStudy, toSnageeFormat

## Examples

```
# Get the list of genes
geneList = getCC()$g;
# Create a random data set
d=list(genes=geneList, data=matrix(rnorm(length(geneList)*50),ncol=50));
# And calculate the quality of the samples (they are all about the same)
qualSample(d);
```

qualStudy *Quality of a study*

## Description

Calculate the quality of a study.

## Usage

```
qualStudy(d,mode="complete",cc=NULL,disattenuate=TRUE)
```

## Arguments

| | |
|---|---|
| d | The study data. If an Eset, toSnageeFormat is called on it. Otherwise, must be a list with fields 'genes' containing the vector of gene IDs (from NCBI's Gene DB) and 'data' containing the actual data. |
| mode | Which gene-gene correlation matrix should be used. Can be 'complete' (using all platforms) or 'woAffy' (without the Affy platforms). |
| cc | Can be used if wishing to use a custom gene-gene correlation matrix. Must be a list with fields 'g' containing the gene IDs and 'cc' containing the (upper triangular part of the) correlations. |
| disattenuate | Should the qualities be disattenuated? |

## Details

The function calculates the quality of a study. The numerical values of the study (the 'data' field) should be in log-scale, and normalized. It is recommended to used medpolish on the data.

Each gene should only appear once in the gene list. Duplicated genes must be merged before using the function.

The mode `'woAffy'` may be useful to compare Affymetrix to not Affymetrix studies. As the median gene correlation matrix was calculated with a majority of Affymetrix platforms, those platforms tend to be given higher quality than the others with the `'complete'` mode, which may be misleading.

## See Also

SNAGEE, qualSample, linktoSnageeFormat

## Examples

```
# Get the list of genes
geneList = getCC()$g;
# Create a random data set
d=list(genes=geneList, data=matrix(rnorm(length(geneList)*50),ncol=50));
# And calculate its quality (it's going to be close to 0)
qualStudy(d, disattenuate=FALSE);
```

`toSnageeFormat` *Turns an Eset into a list*

## Description

Turns an Eset into a list usable by SNAGEE.

## Usage

```
toSnageeFormat(data)
```

## Arguments

| | |
|---|---|
| `data` | An Eset. If already a list, leaves it as it is. |

## Details

The function turns an Eset into a list usable by SNAGEE. Gene ID annotations are found using the annotation slot of the Eset, and the related annotation DB. If no annotation DB can be found, gives an error.

In addition, features with identical gene IDs are averaged, and the data are medpolished.

## See Also

SNAGEE, qualStudy, qualSample

## Examples

```
# Get the list of genes
geneList = getCC()$g;
# Create a random data set
d=list(genes=geneList, data=matrix(rnorm(length(geneList)*50),ncol=50));
# And calculate its quality (it's going to be close to 0)
qualStudy(d, disattenuate=FALSE);
```

# Index