

Package ‘a4Preproc’

May 27, 2024

Type Package

Title Automated Affymetrix Array Analysis Preprocessing Package

Version 1.52.0

Date 2020-10-14

Description Utility functions to pre-
process data for the Automated Affymetrix Array Analysis set of packages.

Imports BiocGenerics, Biobase

Suggests ALL, hgu95av2.db, knitr, rmarkdown

License GPL-3

biocViews Microarray, Preprocessing

RoxygenNote 7.1.1

VignetteBuilder knitr

git_url <https://git.bioconductor.org/packages/a4Preproc>

git_branch RELEASE_3_19

git_last_commit cfc6b6f

git_last_commit_date 2024-04-30

Repository Bioconductor 3.19

Date/Publication 2024-05-27

Author Willem Talloen [aut],
Tobias Verbeke [aut],
Laure Cougnaud [cre]

Maintainer Laure Cougnaud <laure.cougnaud@openanalytics.eu>

Contents

| | |
|-----------------------|---|
| addGeneInfo | 2 |
| Index | 4 |

addGeneInfo

Utility Function to Add Annotation to existing ExpressionSet Objects

Description

Adds appropriate featureData, similar to the metadata added in the pipeline script, to the ExpressionSet object.

Usage

```
addGeneInfo(eset, annotationLibrary = NULL)
```

Arguments

| | |
|-------------------|---|
| eset | ExpressionSet object for to which one wants to add additional annotation information |
| annotationLibrary | Annotation Library to use. Must be specified when working with Entrez gene IDs. In this case, one can make use of the JnJ annotation packages such as hgu133plus2hsentrezgJnJ. If not specified, the annotation of the package will be automatically requested with <code>annotation()</code> of the expressionSet object eset and then Affymetrix probe set IDs are expected in featureNames |

Details

Slots of featureData(a4ALL) are

- Entrez ID~: Entrez ID as retrieved from annotation package
- Ensembl ID~: Ensembl ID as retrieved from annotation package
- Gene Symbol~: Gene symbol as retrieved from annotation package
- Description~: Description as retrieved from annotation package

Value

a new ExpressionSet object with the additional information stored as feature data

Note

One should always use subscripting of featureData by column name (e.g. `featureData(a4ALL)$`Entrez ID``; as the pipeline ExpressionSets have one additional column compared to the ExpressionSet objects produced by addGeneInfo, i.e. column 2 of the pipeline ExpressionSets corresponds to column one of an addGeneInfo ExpressionSet.

Author(s)

Tobias Verbeke, Steven Osselaer

Examples

```
library(ALL)
data(ALL)
a4ALL <- addGeneInfo(ALL)
head(featureData(a4ALL)$`Entrez ID`)
```

Index

* **manip**

addGeneInfo, [2](#)

addGeneInfo, [2](#)