

# Package ‘agilp’

February 14, 2012

**Type** Package

**Title** Extracting and preprocessing Agilent express arrays

**Version** 1.0

**Date** 2010-03-12

**Author** Benny Chain <b.chain@ucl.ac.uk>

**Maintainer** Benny Chain <b.chain@ucl.ac.uk>

**Description** The current version contains a single function, AAProcess. AAProcess extracts the raw median signals from green and red channels of Agilent scanner array files. It replaces values for replicate probes by the mean of all replicates, and (optionally) removes control probes. It log<sub>2</sub> transforms data, and outputs txt files containing the red and green channel data separately. It then normalises each set of data against a baseline file (typically median or mean of a large set of arrays, using LOESS regression). Finally outputs the normalised data. The normalisation step can be optionally omitted.

**License** GPL (>= 2)

**LazyLoad** no

**Repository** CRAN

**Date/Publication** 2010-03-15 07:59:33

## R topics documented:

agilp-package . . . . .	2
AAProcess . . . . .	2

<b>Index</b>	<b>5</b>
--------------	----------

agilp-package      *The package contains functions for processing and analysing Agilent  
express arrays*

---

### Description

In this release the function contains a single function, AAPProcess. For a description of this function see ?AAPProcess.

### Details

Package:      agilp  
Type:          Package  
Version:      1.0-0  
Date:          2010-03-05  
License:      GPL-2  
LazyLoad:    yes

See above

### Author(s)

Benny Chain b.chain@ucl.ac.uk

Maintainer: Benny Chain b.chain@ucl.ac.uk>

### References

See documentation under AAPProcess function

---

AAPProcess      *Extract and preprocess Agilent expression array data*

---

### Description

AAPProcess extracts the raw median signals from green and red channels of Agilent scanner array files. It replaces values for replicate probes by the mean of all replicates, and (optionally) removes control probes .

It log2 transforms data , and outputs txt files containing the red and green channel data separately. It then normalises each set of data against a baseline file (typically median or mean of a large set of arrays, using LOESS regression. Finally outputs the normalised data. The normalisation step can be optionally omitted.

**Usage**

```
AAProcess(input, output, baseline, s = 9, norm = TRUE)
```

**Arguments**

input	full path of directory where input data files are found
output	full path of directory where output data files will be written
baseline	full path of file to be used as baseline for normalisation
s	the number of lines occupied by the header in the scanner output data files. The default is 9
norm	logical variable determining whether or not to carry out normalisation. Default is TRUE

**Details**

The input directory must have ONLY the data files to be analysed. The program checks that all entries are numeric and gives an error message if they are not.

The normalisation code requires a user supplied baseline file. A suitable file for the Agilent 44k human expression arrays is provided in the parent `agilp` folder of the package (named `baseline.txt`). It contains median values for each probe calculated from 280 expression arrays. If the package is used on different Agilent arrays, or if the user prefers to create their own baseline file, run `AAProcess` with `norm = FALSE`. Then use the raw output `preprocessg` and `preprocessr` objects to create a new baseline file containing the median values for each probe. Note that the baseline file must have rownames which match the probe names of the data files to be normalised, or the function will generate an error message. Save the file as a tab delimited table (extension `.txt`).

**Value**

rawdata files	The function outputs a set of txt files containing the raw data for each channel in directory defined by <code>output</code> . The raw data files are given the same name as the input data files with the prefix <code>rawg</code> and <code>rawr</code> for green and red channel respectively. Note that the rawdata files will contain values for all probes on array, while normalised data files will contain only those probes for which baseline values are supplied in file <code>baseline</code>
ndata	Optionally a set of files giving the normalised data. The normalised data files are given the same name as the input data files with the prefix <code>ng</code> and <code>nr</code> for green and red channel respectively.
datag and datar	Matrices, whose columns correspond to the individual arrays, and whose rows are the log <sub>2</sub> transformed raw probe intensities (green and red channel respectively).
normg and normr	Similar matrices with normalised data. These matrices can be used for further data analysis/visualisation as required.

**Author(s)**

Benny Chain; b.chain@ucl.ac.uk

**References**

From analysis of error and reproducibility to a pipeline for data processing of Agilent oligonucleotide expression arrays.

Benjamin Chain, Helen Bowen, John Hammond, Wilfried Posch Jane Rasaiyaah, Jhen Tsang and Mahdad Noursadeghi. BMC Bioinformatics Submitted

**Examples**

## Not run:

Make a new directory C:\input , and then copy the sample datafiles (Data1.txt and Data2.txt) from agilp package directory

Make a new directory C:\baseline and copy the file baseline.txt from agilp package directory to this new directory.

Make a new directory C:\output.

Note the path names are defined using double backslashes in the function call, or R will interpret the backslashes as escape characters.

Also, the pathnames must be enclosed in double quotes.

```
AAPProcess(input = "C:\\input\\", output = "C:\\output\\", baseline = "C:\\baseline\\baseline.txt")
```

The number of lines header to be skipped from scanner array files can be changed from the default value of 9 by setting

The normalisation can be omitted by setting norm = FALSE

```
AAPProcess(input = "C:\\input\\", output = "C:\\output\\", baseline = "C:\\baseline\\baseline.txt", s = 12, norm = F
```

```
## End(Not run)
```

# Index

\*Topic **data**

AAProcess, [2](#)

\*Topic **manip**

AAProcess, [2](#)

\*Topic **package**

agilp-package, [2](#)

AAProcess, [2](#)

agilp (agilp-package), [2](#)

agilp-package, [2](#)