

yaqcaffy

April 19, 2010

getAllInt

Get the GCOS values for a given spike probe

Description

This function retrieves the expression intensities provided by the object of class "YAQCStats" for the probe which names are compatible with the given pattern and return their mean value.

Usage

```
getAllInt (YAQCStatsObject, pattern)
```

Arguments

YAQCStatsObject an object of type "YAQCStats"
pattern a pattern used to select the probe names to be used

Value

An object of type "numeric" in which the mean expression intensities of each array are given.

Author(s)

Laurent Gatto

Examples

```
## load a dataset  
library(affydata)  
data(Dilution)  
## perform quality control  
qc <- yaqc(Dilution)  
## get intensities for the biob  
## spikes probe sets  
getAllInt(qc, "biob")
```

`getBioProbes`*Get the names of the Bio spike probes on the array*

Description

This function returns all the AFFX-Bio probes names that are located on the given GeneChip.

Usage

```
getBioProbes(affyBatchObject)
```

Arguments

```
affyBatchObject  
  An object of type "AffyBatch"
```

Value

An object of type "character" with all the Affymetrix Bio probe names.

Author(s)

Laurent Gatto

See Also

`getSpikeProbes`, `getRatioProbes`

Examples

```
## load a dataset  
library(affydata)  
data(Dilution)  
getBioProbes(Dilution)
```

`getOutliers`*Get outliers for the different YAQCStatsObject slots*

Description

This function retrieves the outliers for the different quality control metrics stored in a YAQCStatsObject. Outliers are defined as being outside of the mean ± 2 stdev range or $\text{mean}/2$, $\text{mean} * 1.5$ for the scale factor.

Usage

```
getOutliers(YAQCStatsObject, slot)
```

Arguments

YAQCStatsObject	an object of type "YAQCStats"
slot	an object of type <code>string</code> describing the slot for which the outliers should be retrieved (see details for possible slot strings)

Details

The slot strings that can be used are:

scale factor "sfs"

average background "avbg"

average noise "avns"

percentage present "pp"

β -actin 3'/5' ratio "actin"

GAPDH 3'/5' ratio "gapdh"

internal bioB control "biob"

internal bioC control "bioc"

internal bioD control "biod"

Dap spike control "dap"

Thr spike control "thr"

Phe spike control "phe"

Lys spike control "lys"

Value

An object of type "numeric" giving the outliers names and values

Author(s)

Laurent Gatto

Examples

```
## load data
library(MAQCsubsetAFX)
data(refA)
## create the yaqc object
qobj <- yaqc(refA[, 3:5])
## get outliers for the scale factor
getOutliers(qobj, "sfs")
```

getQCRatios *Compute qc probe ratios using GCOS intensity values*

Description

This function computes the 3'/5' ratios of the GAPDh and β -actin qc probes using the GCOS intensity values.

Usage

```
getQCRatios(YAQCStatsObject)
```

Arguments

YAQCStatsObject
an object of class YAQCStats

Value

An object of type "matrix" with two qc ratios per array.

Author(s)

Laurent Gatto

See Also

getRatioProbes

Examples

```
## load a dataset
library(affydata)
data(Dilution)
## create yaqc object
qobj <- yaqc(Dilution)
getQCRatios(qobj)
```

getRatioProbes *Get the names of degradation control probes on the array*

Description

This function returns the probes names used for degradation control that are located on the given GeneChip.

Usage

```
getRatioProbes(object)
```

Arguments

object An object of class "AffyBatch" or "ExpressionSet"

Value

An object of type "character" with all the Affymetrix degradation control probe names.

Author(s)

Laurent Gatto

See Also

getSpikeProbes, getBioProbes

Examples

```
library(yaqcaffy)
## load a dataset
library(affydata)
data(Dilution)
getRatioProbes(Dilution)
```

getSpikeProbes *Get the names of all spike probes on the array*

Description

This function returns all the spike probes (i.e. BioB-3', BioD-5', Lys-3, ...) that are located on the given GeneChip.

Usage

```
getSpikeProbes(object)
```

Arguments

object An object of type AffyBatch or ExpressionSet.

Value

An object of class character containing all Affymetrics spike probe names.

Author(s)

Laurent Gatto

See Also

getBioProbes, getRatiosProbes

Examples

```
## load a dataset
library(affydata)
data(Dilution)
getSpikeProbes(Dilution)
```

morespikes	<i>The name of more spike probes</i>
------------	--------------------------------------

Description

The name of more spike probes for each array (e.g. biob-3, biob-5, etc..)

Format

A CSV file, with a column for each spike.

See Also

spikes

reprodPlot	<i>Plot human whole genome GeneChips reproductibility</i>
------------	---

Description

Compares Affymetrix Human Genome U133 Plus 2.0 Arrays to a subset of the MAQC arrays for a RNA reference.

Usage

```
reprodPlot(userAffyBatchObject, ref,
           normalize=c("rma", "gcrma", "mas5", "none"),
           main="MAQC reference reproductibility",
           cex, ...)
```

Arguments

userAffyBatchObject	a set of Human Genomue U133 Plus 2.0 arrays provided as an AffBatch object,
ref	a string ("refA", "refB", "refC", or "refD") defining the RNA reference to compare the userAffyBatchObject to,
normalize	a string defining the algorithm used for data normalization: <code>rma</code> (default) for RMA (as implemented in the affy library), <code>gcrma</code> for GCRMA (as implemented in the <code>gcrma</code> library), <code>mas5</code> for MAS5 (as implemented in the affy library) or 'none' for no normalization,
main	an overall title for the plot,
cex	size of text on the plot,
...	other arguments.

Details

The plot shows all the pairwise scatterplots (plotted with **graphics**'s `smoothScatter` function) with Pearson's correlation factor and MAplots (plotted with **affy**'s `ma.plot` function). The subset of the MAQC arrays are 1 randomly chosen .CEL file out of the 5 replicates for the 6 different test site.

Value

Outputs a graph on the available graphical device

Author(s)

Laurent Gatto

Examples

```
## Not run:
## loading data
library(MAQCsubsetAFX)
data(refB)
d<-refB[,1]
## testing the reproductibility against ref A
reprodPlot(d, "refA", normalize="rma")

## End(Not run)
```

yaqc.affy

Generate QC stats from an AffyBatch object

Description

Generate YAQC metrics for Affymetrix data.

Usage

```
yaqc.affy(object, tgt=100, tau=0.015, logged=FALSE)
```

Arguments

object	a object of type <code>AffyBatch</code> or <code>ExpressionSet</code>
tgt	the target intensity to which the chips should be scaled (used to calculated the GCOS intensities)
tau	used by detection p value
logged	to be used with an <code>ExpressionSet</code> object, defining if the expression intensities are logged

Details

Affymetrix recommend a set of quality control metrics to check the quality of GeneChips expression arrays. This function applies the guidelines described in the Affymetrix Microarray Quality Control Consortium (MAQC) protocols to asses the succes of the hybridization. See the package vignette for more details.

This function takes a raw (unnormalised) `AffyBatch` object or and `ExpressionSet` object. In the first case, it computess GCOS intensity values, MAS expression calls(see `call.exprs`) and other quality-related metrics to generate an `YAQCStats`. If an `ExpressionSet` object is provided, only the β -actin, GAPDH and internal control values are computed.

Value

An `YAQCStats` object describing the input object

Author(s)

Laurent Gatto

Examples

```
## loading data
library(affydata)
data(Dilution)
## qc analysis
qobj <- yaqc(Dilution)
show(qobj)
```

yaqc-methods

Create an YAQCStat object

Description

Create an `YAQCStats` qc object for an `AffyBatch` (or `ExpressionSet`) input

Methods

object="eSet" Create a full `YAQCStats` object for an `AffyBatch` input or a light `YAQCStats` object for am `ExpressionSet` input.

See Also

[yaqc.affy](#)

`yaqc.plot`*Plots a YAQCStats object*

Description

Generates a visual summary of the various Affymetrix QC statistics.

Usage

```
yaqc.plot(YAQCStatsObject, which, ...)
```

Arguments

<code>YAQCStatsObject</code>	an object of class "YAQCStats".
<code>which</code>	which quality metrics should be plotted (all by default).
<code>...</code>	other arguments.

Details

The quality control metrics of the `YAQCStatsObject` are plotted in a serie of graphs with the recommended ranges.

The scale factors are represented through a dot chart and the upper and lower limits are defined with vertical red lines. The other qc metircs are shown using dot plots. For the upper row box plots (average background, average noise, percent present and β -actin and GAPDH ratios, the mean is represented by a dashed red line and the mean +/- 2 stdev by red dotted lines. For the lower box plots, featuring the internal controls, grey boxes defines the mean (middle segment) +/- 2 stdev.

Individual plots can also be generated with the `which` argument: 'sfs' for the scale factor, 'avbg' and 'avns' for the average background and noise, 'pp' for the percentage of present calls, 'gapdh' and 'actin' for the GAPDH and β -actin ratios, 'bio' for the hybridization controls and 'spikes' for the retro-transcription spiked controls.

If the `YAQCStatsObject` has been generated with an Expression Set objects, the scale factors, average noise and background and percent present can not be computed and the respective plots are removed from the final graph.

Author(s)

Laurent Gatto

Examples

```
## load data
library(affydata)
data(Dilution)
## create the yaqc object
## and plot it
qobj <- yaqc(Dilution)
plot(qobj)
```

moreSpikeInProbes *Accessor Methods for YAQCStats class*

Description

The Microarray Quality Control Consortium guidelines and reference datasets are applied to analyse some of the QC metrics recommended by Affymetrix. These guidelines, datasets and QC metrics are discussed in detail in the package vignette.

Each of these functions queries a YAQCStats object to retrieve one of these QC metrics. The YAQCStats class is a subclass of the QCStats object as defined in the **simpleaffy** package. `sfs`, `avbg` and `percent.present` methods access to the `scale.factors`, `average.background` and `percent.present` attributes respectively are defined in the superclass.

Briefly:

`isLog`: Returns a logical saying if the expression intensities are in log2 from.

`target`: Returns the target value (for MAS 5.0 normalization).

`moreSpikeInProbes`: Returns a table of intensities for the internal spike probes.

`gcosProbes`: Returns a table of intensities for GAPDH and β -actin probes.

`avbg`: Returns a vector of the average background levels for each array.

`minbg`: Returns a vector of the minimum background levels for each array.

`percent.present`: Returns a vector listing the percentage of probesets called present on each array.

`sfs`: Returns a vector of scale factors for each array (as produced by the MAS 5.0 algorithm).

`bioCalls`: Returns a table of Present/Marginal/Absent calls for the spike probes.

`arrays`: Returns the names of the arrays in the YAQCStats object.

`plot`: Plots the YAQCStats object (see [yaqc](#) for more details).

Usage

```
isLog(object)
avns(object)
moreSpikeInProbes(object)
gcosProbes(object)
bioCalls(object)
```

Arguments

`object` an object of class "YAQCStats"

Author(s)

Laurnet Gatto

See Also

[yaqc](#), `QCStats`

YAQCStats-class *Class "YAQCStats"*

Description

Holds Quality Control data for a set of Affymetrix arrays

Objects from the Class

Objects can be created by calls of the form `yaqc(object)` where `object` is of class `AffyBatch` of `ExpressionSet`. `YAQCStats` is a subclass of `QCStats` and uses the `scale.factor`, `average.background` and `percent.present`, `arraytype` and `target` attributes of it's subclass.

Slots

`log`: Object of class "logical" that specifies if expression values are in log2 form

`average.noise`: Object of class "numeric" The average noise for the arrays

`morespikes`: Object of class "matrix" More spiked in probes (e.g. `r2biob5`, `r2biob3`,...)

`gcos.probes`: Object of class "matrix" GAPDH and β -actin qc probes (e.g. `gapdh 3,5,M`,...) containing the GCOS values

`bio.calls`: Object of class "matrix" BioB 5',3',M and BioC, BioC 5',3' present/absent/marginal calls

Methods

isLog signature(object = "YAQCStats"): are intensities in log2 form

avns signature(object = "YAQCStats"): average noise

moreSpikeInProbes signature(object = "YAQCStats"): more spike-in probes

gcosProbes signature(object = "YAQCStats"): GCOS values of `gapdh` and `actin` QC probes

bioCalls signature(object = "YAQCStats"): P/M/A calls for BioB, BioC and BioD spikes

arrays signature(object = "YAQCStats"): shows the array names composing the `YAQCStats` object

plot signature(object = "YAQCStats"): visual representation of the qc metrics

summary signature(object = "YAQCStats"): data frame summary of the outliers

show signature(object = "YAQCStats"): displays the content of the object as a data frame

merge signature(object = "YAQCStats"): merges two compatible `YAQCStats` objects, i.e. that have the same values for the `log`, `target` and `arraytype` slots.

arrays signature(object = "YAQCStats"): show the array names of an `YAQCStats` objects.

Author(s)

Laurent Gatto

See Also

QCStats-class

yaqc.summary	<i>Summarizes YAQCStats object outliers</i>
--------------	---

Description

Generates a data frame of QC outliers.

Usage

```
yaqc.summary(YAQCStatsObject, latex)
```

Arguments

YAQCStatsObject

an object of class "YAQCStats".

latex

logical value. If 'TRUE', the data frame is returned as a latex table.

Details

The outliers of the YAQCStatsObject quality control metrics are summarized and returned as a data frame. The internal probe (bioB, bioC and bioD) and poly-A controls (dap, thr, phe and lys) are summarized in two rows.

If latex is set to 'TRUE', the data frame is returned as a latex table (requires the xtable library).

Note: not tested if the YAQCStatsObject has been generated with an Expression Set objects.

Author(s)

Laurent Gatto

Examples

```
## load data
library(affydata)
data(Dilution)
## create the yaqc object
## and plot it
qobj <- yaqc(Dilution)
summary(qobj)
```

Index

*Topic classes

YAQCStats-class, 11

*Topic datasets

morespikes, 6

*Topic methods

yaqc-methods, 8

*Topic misc

getAllInt, 1

getBioProbes, 2

getOutliers, 2

getQCRatios, 4

getRatioProbes, 4

getSpikeProbes, 5

moreSpikeInProbes, 10

reprodPlot, 6

yaqc.affy, 7

yaqc.plot, 9

yaqc.summary, 12

arrays (*moreSpikeInProbes*), 10

arrays, YAQCStats-method
(YAQCStats-class), 11

arrays-methods (YAQCStats-class),
11

avns (*moreSpikeInProbes*), 10

avns, YAQCStats-method
(YAQCStats-class), 11

avns-methods (YAQCStats-class), 11

bioCalls (*moreSpikeInProbes*), 10

bioCalls, YAQCStats-method
(YAQCStats-class), 11

bioCalls-methods
(YAQCStats-class), 11

gcosProbes (*moreSpikeInProbes*), 10

gcosProbes, YAQCStats-method
(YAQCStats-class), 11

gcosProbes-methods
(YAQCStats-class), 11

gcrma, 6

getAllInt, 1

getBioProbes, 2

getOutliers, 2

getQCRatios, 4

getRatioProbes, 4

getSpikeProbes, 5

isLog (*moreSpikeInProbes*), 10

isLog, YAQCStats-method
(YAQCStats-class), 11

isLog-methods (YAQCStats-class),
11

ma.plot, 7

merge, YAQCStats-method
(YAQCStats-class), 11

moreSpikeInProbes, 10

moreSpikeInProbes, YAQCStats-method
(YAQCStats-class), 11

moreSpikeInProbes-methods
(YAQCStats-class), 11

morespikes, 6

plot (*moreSpikeInProbes*), 10

plot, YAQCStats (*yaqc.plot*), 9

plot, YAQCStats, ANY-method
(YAQCStats-class), 11

plot, YAQCStats, missing-method
(*yaqc.plot*), 9

plot, YAQCStats-method
(*yaqc.plot*), 9

reprodPlot, 6

show, YAQCStats-method
(YAQCStats-class), 11

smoothScatter, 7

summary, YAQCStats (*yaqc.summary*),
12

summary, YAQCStats, ANY-method
(YAQCStats-class), 11

summary, YAQCStats, missing-method
(*yaqc.summary*), 12

summary, YAQCStats-method
(*yaqc.summary*), 12

target, YAQCStats-method
(YAQCStats-class), 11

`target-methods (YAQCStats-class)`,
[11](#)

`yaqc`, [10](#)

`yaqc (yaqc.affy)`, [7](#)

`yaqc, eSet-method (yaqc-methods)`, [8](#)

`yaqc-methods`, [8](#)

`yaqc.affy`, [7](#), [8](#)

`yaqc.plot`, [9](#)

`yaqc.summary`, [12](#)

`YAQCStats-class`, [11](#)