

Package ‘AssayCorrector’

December 23, 2016

Type Package

Title AssayCorrector allows one to detect and eliminate assay-specific and plate-specific spatial biases (both additive and multiplicative) in experimental HTS and HCS assays

Version 1.1.1

Date 2016-12-23

Author Bogdan Mazoure (McGill University, Department of Computer Science), Iurie Caraus (UQAM, Departement d'Informatique), Robert Nadon (McGill University, Department of Human Genetics and Genome Quebec Innovation Center) and Vladimir Makarenkov (UQAM, Departement d'Informatique)

Maintainer Bogdan Mazoure <bogdan.mazoure@gmail.com>

Description (1) AssayCorrector allows one to detect plate-specific spatial bias by identifying rows and columns of all plates of the assay affected by this bias (following the results of the Mann-Whitney U test) as well as assay-specific spatial bias by identifying well locations (i.e., well positions scanned across all plates of a given assay) affected by this bias (also following the results of the Mann-Whitney U test);(2) AssayCorrector allows one to correct plate-specific spatial bias using either the additive or multiplicative PMP (Partial Mean Polish) method (the most appropriate spatial bias model can be either specified by the user or determined by the program following the results of the Kolmogorov-Smirnov two-sample test) to correct the assay measurements as well as to correct assay-specific spatial bias by carrying out robust Z-scores within each plate of the assay and then traditional Z-scores across well locations.

Depends R (>= 3.3.0),
RColorBrewer,
lattice

License GPL-3

LazyData TRUE

RoxygenNote 5.0.1

R topics documented:

correct_bias	2
create_assay	2
detect_bias	3
plot.assay	4
print.assay	4
Index	5

correct_bias	<i>Correct the bias present in the assay, previously detected by the detect_bias() method</i>
--------------	---

Description

correct_bias() (1) uses either the additive or multiplicative PMP (Partial Mean Polish) methods (the most appropriate spatial bias model can be either specified or determined by the program following the results of the Kolmogorov-Smirnov two-sample test) to correct the assay measurements if the plate-specific correction is specified; (2) carries out the assay-specific correction if specified.

Usage

```
correct_bias(assay, method=NULL, alpha=0.05, type="PA")
```

Arguments

assay	The assay to be corrected. Has to be an assay object.
method	NULL:autodetect (default), 1:additive, 2:multiplicative
alpha	Significance level threshold (defaults to 0.05)
type	P:plate-specific, A:assay-specific, PA:plate then assay-specific, AP:assay then plate-specific

Value

The corrected assay (assay object)

Examples

```
m<-readRDS(gzcon(url(
  'https://github.com/ArtificialBreeze/AssayCorrector/blob/master/examples/8x12_raw.Rda?raw=true')))
assay<-create_assay(m)
detected<-detect_bias(assay)
corrected<-correct_bias(detected, method=2)
```

create_assay	<i>Create a new assay</i>
--------------	---------------------------

Description

create_assay makes a new object of class assay. You should pass this object to detect_bias() and correct_bias() methods

Usage

```
assay<-create_assay(m, ctrl=NA)
```

Arguments

<code>m</code>	The assay you want to be corrected
<code>ctrl</code>	An optional boolean array of the same dimensions as <code>m</code> . Each entry is 1 if the well is a control well, 0 otherwise. All control wells are excluded from all computations

Value

`assay` The created assay object. It contains the following fields:

`n` The HTS matrix of raw measurements

`ctrl` The binary matrix of control wells

`biasPositions` The binary matrix where 1:well is biased, 0:well is unbiased, as suggested by Mann-Whitney test

`mCorrected` The HTS matrix of corrected measurements, initialized to a zero array, and subsequently storing the corrected version of `m` via `correct_bias()`

`biasType` Vector of length `p`, where `p` is the number of plates. It tell, for each plate of the assay, A:Additive trend, M:Multiplicative trend, U:Undetermined trend and C>Error-free plate.

Examples

```
# Fictive 8x12x5 assay
m<-readRDS(gzcon(url(
  'https://github.com/ArtificialBreeze/AssayCorrector/blob/master/examples/8x12_raw.Rda?raw=true'))))
assay<-create_assay(m)
# Plate 7 taken from Carralot et al. 2012
m<-readRDS(gzcon(url(
  'https://github.com/ArtificialBreeze/AssayCorrector/blob/master/examples/Plate7_raw.Rda?raw=true'))))
assay<-create_assay(m)
```

detect_bias

*Detect the type of bias present in the assay***Description**

`detect` (1) identifies rows and columns of all plates of the assay affected by spatial bias (following the results of the Mann-Whitney U test); (2) identifies well locations (i.e., well positions scanned across all plates of a given assay) affected by spatial bias (also following the results of the Mann-Whitney U test).

Usage

```
detect_bias(assay, alpha=0.01, type="P")
```

Arguments

<code>assay</code>	The assay to be corrected. Has to be an assay object.
<code>alpha</code>	Significance level threshold (defaults to 0.05)
<code>type</code>	P:plate-specific, A:assay-specific, PA:plate then assay-specific, AP:assay then plate-specific

Value

The corrected assay (assay object)

Examples

```
m<-readRDS(gzcon(url(
  'https://github.com/ArtificialBreeze/AssayCorrector/blob/master/examples/8x12_raw.Rda?raw=true'))))
assay<-create_assay(m)
detected<-detect_bias(assay)
```

plot.assay	<i>Plot assay plate-wise</i>
------------	------------------------------

Description

plot.assay plots a hit map of the assay (only one plate at a time)

Usage

```
plot(assay,plate=2,type="P")
```

Arguments

assay	The assay you want to plot
plate	The plate number
type	Either "R" - raw assay, "C" - corrected assay (if it exists) or "P" - spatial bias position

print.assay	<i>Print assay summary</i>
-------------	----------------------------

Description

print.assay simply prints a summary of the HTS assay

Usage

```
print(assay,plate=2)
```

Arguments

assay	The assay you want to print
plate	The plate number

Index

`correct_bias`, [2](#)

`create_assay`, [2](#)

`detect_bias`, [3](#)

`plot.assay`, [4](#)

`print.assay`, [4](#)