Package 'MDimNormn'

January 20, 2025

Title Multi-Dimensional MA Normalization for Plate Effect

Version 0.8.0

Author Mun-Gwan Hong

Maintainer Mun-Gwan Hong <mun-gwan.hong@scilifelab.se>

Depends R (>= 3.2.0)

Description Normalize data to minimize the difference between sample plates (batch effects). For given data in a matrix and grouping variable (or plate), the function 'normn_MA' normalizes the data on MA coordinates. More details are in the citation. The primary method is 'Multi-MA'. Other fitting functions on MA coordinates can also be employed e.g. loess.

License GPL-3

LazyData true

LazyLoad yes

Encoding UTF-8

NeedsCompilation no

Repository CRAN

Date/Publication 2015-08-12 20:12:49

Contents

normn MA *Multi-dimensional MA normalization for plate effect*

Description

Normalize data to minimize the difference among the subgroups of the samples generated by experimental factor such as multiple plates (batch effects)
- the primary method is Multi-MA, but other fitting function, f in manuscript (e.g. loess) is available, too.
This method is based on the assumptions stated below

1. The geometric mean value of the samples in each subgroup (or plate) for a single target is ideally same as those from the other subgroups.
2. The subgroup (or plate) effects that influence those mean values for multiple observed targets are dependent on the values themselves. (intensity dependent effects)

Usage

```
normn_MA(mD, expGroup, represent_FUN= function(x) mean(x, na.rm= T),
        fitting_FUN= NULL, isLog= TRUE)
```

Arguments

mD	a matrix of measured values in which columns are the measured molecules and rows are samples
expGroup	a vector of experimental grouping variable such as plate. The length of code-expGroup must be same as the number of rows of mD.
represent_FUN	a function that computes representative values for each experimental group (e.g. plate). The default is mean ignoring any NA
fitting_FUN	NULL or a function that fits to data in MA-coordinates. If it is NULL as the default, 'Multi-MA' method is employed. If a function is used, two arguments of m_j and A are required, which are $\mathbf{m}_j$ coordinate in M_d and A coordinate, respectively.
isLog	TRUE or FALSE, if the normalization should be conducted after log-transformation. The affinity proteomics data from suspension bead arrays is recommended to be normalized using the default, isLog = TRUE.

Value

The data after normalization in a matrix

Author(s)

Mun-Gwan Hong <<mun-gwan.hong@scilifelab.se>>

References

Hong M-G, Lee W, Pawitan Y, Schwenk JM (201?) Multi-dimensional normalization of plate effects for multiplexed applications *unpublished*

Examples

```
data(sba)
B <- normn_MA(sba$X, sba$plate) # Multi-MA normalization

# MA-loess normalization
B <- normn_MA(sba$X, sba$plate, fitting_FUN= function(m_j, A) loess(m_j ~ A)$fitted)

# weighted linear regression normalization
B <- normn_MA(sba$X, sba$plate, fitting_FUN= function(m_j, A) {
beta <- lm(m_j ~ A, weights= 1/A)$coefficients
beta[1] + beta[2] * A
})

# robust linear regression normalization
if(any(search() == "package:MASS")) { # excutable only when MASS package was loaded.
B <- normn_MA(sba$X, sba$plate, fitting_FUN= function(m_j, A) {
beta <- rlm(m_j ~ A, maxit= 100)$coefficients
beta[1] + beta[2] * A
})
}
```

SBA *Artificially generated data alike that of Suspension bead arrays*

Description

The data that has similarity to Suspension bead arrays data.

Usage

```
data(sba)
```

Format

A list that consists of `"plate"` which is a `factor` of plate number, `"X"` that contains measured values where columns are targets and rows are samples (or observations).

Examples

```
data(sba)

# plot to check difference of geometric mean of every target between plates
sba_gm <- by(sba$X, sba$plate, apply, 2, function(x) exp(mean(log(x))))
par(mfrow= c(2, 3))
```

```
apply(combn(4, 2), 2, function(ea) {
plot(sba_gm[[ea[1]]], sba_gm[[ea[2]]], xlab= names(sba_gm)[ea[1]],
     ylab= names(sba_gm)[ea[2]], log= "xy", asp= 1)
abline(0, 1, col= "cadetblue")
})

# show first 10 observations in plate 1 and plate 2
print(sba$X[c(1:10, 97:106), 1:10])
```

Index