Package 'rama'

June 20, 2021

Type Package

Title Robust Analysis of MicroArrays

Version 1.66.0

Author Raphael Gottardo

Description Robust estimation of cDNA microarray intensities with replicates. The package uses a Bayesian hierarchical model for the robust estimation. Outliers are modeled explicitly using a t-distribution, and the model also addresses classical issues such as design effects, normalization, transformation, and nonconstant variance.

Maintainer Raphael Gottardo <raph@stat.ubc.ca>

Depends R(>= 2.5.0)

License GPL (>= 2)

biocViews Microarray, TwoChannel, QualityControl, Preprocessing

git_url https://git.bioconductor.org/packages/rama

git_branch RELEASE_3_13

git_last_commit 8d29b43

git_last_commit_date 2021-05-19

Date/Publication 2021-06-20

R topics documented:

arrange.row *Reorder a dataset by increasing row order.*

Description

The functions could be used to reorder a dataset to make sure that all the genes are in the same row before fitting any model. The `arrange.row` function is also used by the `weight.plot` function to map all the genes to their position on the slide.

Usage

```
arrange.row(data)
```

Arguments

`data`	A dataset containing the row indices in the first column and the column indices in the second column. The row indices should all be distinct. All indices should start at zero!

Value

The ordered dataset.

Author(s)

Raphael Gottardo

References

Robust Estimation of cDNA Microarray Intensities with Replicates Raphael Gottardo, Adrian E. Raftery, Ka Yee Yeung, and Roger Bumgarner Department of Statistics, University of Washington, Box 354322, Seattle, WA 98195-4322

See Also

`weight.plot`

Examples

```
data(hiv)
### Put the indices in the first two columns and
### reorder the first 4 replicates
new.data<-cbind(hiv[,9:10],hiv[,1:4])
ordered.data<-arrange.row(new.data)
```

`est.shift` *Estimate the shift used in the log transformation*

Description

Estimate the shift in the log transformation when fitting the Hierarchical model as in bayes.rob.

Usage

```
est.shift(sample1,sample2,B=1000,min.iter=0,batch=10,mcmc.obj=NULL,dye.swap=FALSE,nb.col1=NULL,all.
```

Arguments

`sample1`	The matrix of intensity from the sample 1. Each row corresponds to a different gene.
`sample2`	The matrix of intensity from the sample 2. Each row corresponds to a different gene.
`B`	The number of iteration used the MCMC algorithm.
`min.iter`	The length of the burn-in period in the MCMC algorithm.`min.iter` should be less than B.
`batch`	The thinning value to be used in the MCMC. Only every `batch`-th iteration will be stored.
`mcmc.obj`	An object of type `mcmc.shift`, as returned by `est.shift`. If no `mcmc.obj`, the MCMC is initialized to the least squares estimates.
`dye.swap`	A logical value indicating if the experiment was a dye swap experiment.
`nb.col1`	An integer value correspinding to the number of arrays (columns) in the first group of the dye swap experiment. In other words, the number of replicates before the dyes have been swaped.
`all.out`	A logical value indicating if all the parameters should be outputted. If `all.out` is FALSE, only the posterior mean is outputted. This could be used to save memory.
`verbose`	A logical value indicating if the current MCMC iteration number should be printed out.

Details

The estimation is done by fitting the same model (as in fit.model) with constant variance, Gaussian errors and a prior for the shift. The main purpose of this function is to estimate the shift in the log transformation. Parameter estimation is carried out using Markov Chain Monte Carlo. The shift is estimated with the posterior mean.

Value

An object of type mcmc.est containing the sampled values from the posterior distribution.

mu	A vector containing the sampled values from mu, the baseline intensity.
alpha2	A vector containing the sampled values from alpha2, the sample effect.
beta2	A vector containing the sampled values from beta2, the dye effect.
delta22	A vector containing the sampled values from delta_22, the dye*sample interaction.
eta	A matrix, each row contains the sampled values from the corresponding array effect.
gamma1	A matrix, each row contains the sampled values from the corresponding gene effect in sample 1.
gamma2	A matrix, each row contains the sampled values from the corresponding gene effect in sample 1.
lambda.gamma1	A vector containing the sampled values for the precision of the gene effect prior in sample 1.
lambda.gamma2	A vector containing the sampled values for the precision of the gene effect prior in sample 2.
rho	A vector containing the sampled values from between sample correlation coefficient rho
lambda_eps1	A vector containing the sampled values from the gene precision in sample 1.
lambda_eps2	A vector containing the sampled values from the gene precision in sample 2.
shift	A vector containing the sampled values from the shift.

Author(s)

Raphael Gottardo

References

Robust Estimation of cDNA Microarray Intensities with Replicates Raphael Gottardo, Adrian E. Raftery, Ka Yee Yeung, and Roger Bumgarner Department of Statistics, University of Washington, Box 354322, Seattle, WA 98195-4322

See Also

fit.model

Examples

```
data(hiv)
### Initialize the proposals
mcmc.hiv<-est.shift(hiv[1:10,c(1:4)],hiv[1:10,c(5:8)],B=2000,min.iter=000,batch=1,mcmc.obj=NULL,dye.swap=TRUE,
```

fit.model *Robust estimation of microarray intensities with replicates*

Description

Estimate the log transformed intensities of each sample of a replicated microarray experiment. The estimation is done via Hiearchical Bayesian Modeling.

Usage

```
fit.model(sample1,sample2,B=1000,min.iter=0,batch=10,shift=NULL,mcmc.obj=NULL,dye.swap=FALSE,nb.col
verbose=FALSE)
```

Arguments

sample1	The matrix of intensity from the sample 1. Each row corresponds to a different gene.
sample2	The matrix of intensity from the sample 2. Each row corresponds to a different gene.
B	The number of iteration used the MCMC algorithm.
min.iter	The length of the burn-in period in the MCMC algorithm.min.iter should be less than B.
batch	The thinning value to be used in the MCMC. Only every batch-th iteration will be stored.
mcmc.obj	An object of type mcmc, as returned by fit.model. mcmc.obj is used to initialized the MCMC. If no mcmc.obj, the MCMC is initialized to the least squares estimates.
shift	The shift to be used in the log transformation. If shift=NULL is specified (default), it is estimated using est.shift
dye.swap	A logical value indicating if the experiment was a dye swap experiment.
nb.col1	An integer value corresponding to the number of arrays (columns) in the first group of the dye swap experiment. In other words, the number of replicates before the dyes have been swaped.
all.out	A logical value indicating if all the parameters should be outputted. If all.out is FALSE, only the posterior mean is outputted. This could be used to save memory.
ci	A number between 0 and 1 corresponding to the level used when computing log ratio credible intervals. If all.out is FALSE, this option is ignored.
verbose	A logical value indicating if the current MCMC iteration number should be printed out.

Details

The function fits a hierarchical Bayesian model for robust estimation of cDNA microarray intensities. Our model addresses classical issues such as design effects, normalization and transformation. Outliers are modeled explicitly using a t-distribution. Parameter estimation is carried out using Markov Chain Monte Carlo.

Value

An object of type `mcmc` containing the sampled values from the posterior distribution.

`mu`	A vector containing the sampled values from `mu`, the baseline intensity.
`alpha2`	A vector containing the sampled values from `alpha2`, the sample effect.
`beta2`	A vector containing the sampled values from `beta2`, the dye effect.
`delta22`	A vector containing the sampled values from `delta_22`, the dye*sample interaction.
`eta`	A matrix, each row contains the sampled values from the corresponding array effect.
`gamma1`	A matrix, each row contains the sampled values from the corresponding gene effect in sample 1.
`gamma2`	A matrix, each row contains the sampled values from the corresponding gene effect in sample 1.
`q.low`	A vector containing the lower bounds for the log ratio credible intervals, i.e. the credible intervals for gamma1-gamma2.
`q.up`	A vector containing the upper bounds for the log ratio credible intervals, i.e. the credible intervals for gamma1-gamma2.
`lambda.gamma1`	A vector containing the sampled values for the precision of the gene effect prior in sample 1.
`lambda.gamma2`	A vector containing the sampled values for the precision of the gene effect prior in sample 2.
`rho`	A vector containing the sampled values from between sample correlation coefficient `rho`
`lambda_eps1`	A matrix, each row contains the sampled values from the corresponding gene precision in sample 1.
`lambda_eps2`	A matrix, each row contains the sampled values from the corresponding gene precision in sample 2.
`a.eps`	A vector containing the sampled values for the mean of the prior of the genes precision.
`b.eps`	A vector containing the sampled values for the variance of the prior of the genes precision.
`w`	A matrix, each element (i,j) correspond to the posterior mean of the sampled weights of replicate j in gene i.To save memory, we only store the posterior means of the weigths.
`shift`	The value of the shift.

Author(s)

Raphael Gottardo

References

Robust Estimation of cDNA Microarray Intensities with Replicates Raphael Gottardo, Adrian E. Raftery, Ka Yee Yeung, and Roger Bumgarner Department of Statistics, University of Washington, Box 354322, Seattle, WA 98195-4322

See Also

`est.shift`

Examples

```
data(hiv)
mcmc.hiv<-fit.model(hiv[1:10,c(1:4)],hiv[1:10,c(5:8)],B=2000,min.iter=000,batch=1,shift=30,mcmc.obj=NULL,dye.s
```

---

hiv — *Cellular gene expression upon human immunodeficiency virus type 1 infection of CD4+-T-Cell lines*

Description

This data set consists of 4 experiments using the same RNA preparation on 4 different slides. The expression levels of ~7000 cellular RNA transcripts were assessed in CD4-T-cell lines at time $t=24$ hour after infection with HIV virus type 1. The first 4 columns correspond to the first treatment state (hiv infected). The second four represent the control state. The experiment is a balanced dye swap experiment. Finally, the last two columns contain the row and column positions of each gene on the array (slide).

Usage

```
data(hiv)
```

Source

http://expression.microslu.washington.edu/expression/vantwoutjvi2002.html

References

van't Wout, A. B. ,Lehrma, G. K., Mikheeva, S. A., O'Keeffe, G. C., Katze, M. G., Bumgarner, R. E., Geiss, G. K. and Mullins, J. I. Cellular gene expression upon human immunodeficiency virus type 1 infection of CD4+-T-Cell lines Journal of Virology, 2003. 77(2):1392-1402.

`is.row.na` *Test if a matrix contains missing values*

Description

The function returns a vector of logical variables, one for each row of the matrix. The variable is `TRUE` if the row does not contain any missing values and `FAlSE` otherwise.

Usage

```
is.row.na(data)
```

Arguments

`data`	The data matrix.

Value

The vector of logical variable

Author(s)

Raphael Gottardo

See Also

is.na

Examples

```
### Generate a matrix
M<-matrix(rnorm(100),10,10)
M[1,1]<-NA
M[1,2]<- -Inf
M[3,10]<-NA

### Indices of the rows without missing values
ind<-is.row.na(M)

### Submatrix of M with finite values
M.finite<-M[ind,]
```

`ls.effect` *Compute the least squares estimates of the all the effects of the general model.*

Description

Compute the least squares estimates of the all the effects of the general model.

Usage

```
ls.effect(sample1,sample2,dye.swap=FALSE,nb.col1=NULL)
```

Arguments

`sample1`	The matrix of intensity from the sample 1. Each row corresponds to a different gene.
`sample2`	The matrix of intensity from the sample 2. Each row corresponds to a different gene.
`dye.swap`	A logical value indicating if the experiment was a dye swap experiment.
`nb.col1`	An integer value correspinding to the number of arrays (columns) in the first group of the dye swap experiment. In other words, the number of replicates before the dyes have been swaped.

Value

`mu`	The baseline intensity
`alpha2`	The sample effect
`beta2`	The dye effect
`delta22`	The dye*sample interaction
`eta`	The array effects
`gamma1`	The genes effects in sample 1
`gamma2`	The genes effect in sample 2
`M1`	The main effects in sample 1
`M2`	The main effects in sample 2
`R1`	The residuals from the sample 1
`R2`	The residuals from the sample 2

Author(s)

Raphael Gottardo

References

Robust Estimation of cDNA Microarray Intensities with Replicates Raphael Gottardo, Adrian E. Raftery, Ka Yee Yeung, and Roger Bumgarner Department of Statistics, University of Washington, Box 354322, Seattle, WA 98195-4322

See Also

`fit.model`

Examples

```
### Compute the least squares effects on the log scale
data(hiv)
ls.fx<-ls.effect(log2(hiv[,c(1:4)]),log2(hiv[,c(5:8)]),dye.swap=TRUE,nb.col1=2)
```

`mat.mean` *Compute the mean and standard deviation of each row in a data matrix*

Description

This function computes the mean and standard deviation of each row in a data matrix. The source code is written in C. As a consequence, the computation is quite fast.

Usage

```
mat.mean(data)
```

Arguments

`data`	The data matrix.

Value

A matrix, the first columns contain the means, the second the standard deviations.

Author(s)

Raphael Gottardo

See Also

`mean`,`sd`

Examples

```
data(hiv)
sample1<-hiv[,1:4]
ms1<-mat.mean(sample1)
```

`ratio.plot` *Plot the estimated log ratios against the overall intensities*

Description

Plot the estimated log2(gamma1/gamma2) against log2(gamma1*gamma2)/2.

Usage

```
ratio.plot(mcmc.obj,col=1,pch=1)
```

Arguments

`mcmc.obj`	An object of class `mcmc` as returned by `fit.model`
`col`	The color to be used for the symbols
`pch`	The type of symbols to be used.

Value

The graph!

Author(s)

Raphael Gottardo

References

Robust Estimation of cDNA Microarray Intensities with Replicates Raphael Gottardo, Adrian E. Raftery, Ka Yee Yeung, and Roger Bumgarner Department of Statistics, University of Washington, Box 354322, Seattle, WA 98195-4322

See Also

`fit.model`

Examples

```
data(hiv)
### Initialize the proposals
mcmc.hiv<-fit.model(hiv[1:10,c(1:4)],hiv[1:10,c(5:8)],B=2000,min.iter=000,batch=1,shift=30,mcmc.obj=NULL,dye.s
ratio.plot(mcmc.hiv,col=1,pch=1)
```

`weight.plot` *Plot the weights of a given array using the spatial location of the genes on the slide*

Description

Plot the weights of a given array using the spatial location of the genes on the slide.This function is a useful diagnostic tool.

Usage

```
weight.plot(mcmc.obj,coordinate,array=1)
```

Arguments

`mcmc.obj`	An object of class `mcmc` as returned by `fit.model`
`coordinate`	The coordinate of each gene on the corresponding array. The coordinates should be a two column integer valued matrix containing the row indices (column 1) and the column indices (column 2). The row indices should all be distinct. All indices should start at zero!
`array`	An integer corresponding to the array number to be plotted.

Value

The image plot of the weights. A small weight (bright color) correspond to an outlier.

Author(s)

Raphael Gottardo

References

Robust Estimation of cDNA Microarray Intensities with Replicates Raphael Gottardo, Adrian E. Raftery, Ka Yee Yeung, and Roger Bumgarner Department of Statistics, University of Washington, Box 354322, Seattle, WA 98195-4322

See Also

`arrange.row`

Examples

```
data(hiv)
### Initialize the proposals
mcmc.hiv<-fit.model(hiv[1:640,c(1:4)],hiv[1:640,c(5:8)],B=1000,min.iter=500,batch=1,shift=30,mcmc.obj=NULL,dye
weight.plot(mcmc.hiv,hiv[1:640,9:10],array=3)
```

Index