Package ‘EBarrays’

March 28, 2017

Version 2.38.0
Date 2007/09/20
Title Unified Approach for Simultaneous Gene Clustering and Differential Expression Identification
Author Ming Yuan, Michael Newton, Deepayan Sarkar and Christina Kendziorski
Maintainer Ming Yuan <myuan@isye.gatech.edu>
Description EBarrays provides tools for the analysis of replicated/unreplicated microarray data.
Depends R (>= 1.8.0), Biobase, lattice, methods
Imports Biobase, cluster, graphics, grDevices, lattice, methods, stats
License GPL (>= 2)
biocViews Clustering, DifferentialExpression
NeedsCompilation yes

R topics documented:

<table>
<thead>
<tr>
<th>Description</th>
<th>Usage</th>
</tr>
</thead>
<tbody>
<tr>
<td>crit.fun</td>
<td>Find posterior probability threshold to control FDR</td>
</tr>
<tr>
<td>ebarraysFamily-class</td>
<td></td>
</tr>
<tr>
<td>ebplots</td>
<td></td>
</tr>
<tr>
<td>emfit</td>
<td></td>
</tr>
<tr>
<td>gould</td>
<td></td>
</tr>
<tr>
<td>postprob</td>
<td></td>
</tr>
<tr>
<td>utilities</td>
<td></td>
</tr>
</tbody>
</table>

Index 12
ebarraysFamily-class

Class of Families to be used in the EBarrays package

Description

Objects used as family in the emfit function.

The package contains three functions that create such objects for the three most commonly used families, Gamma-Gamma, Lognormal-Normal and Lognormal-Normal with modified variances. Users may create their own families as well.
**Usage**

```
eb.createFamilyGG()
eb.createFamilyLNN()
eb.createFamilyLNNMV()
```

**Details**

The emfit function can potentially fit models corresponding to several different Bayesian conjugate families. This is specified as the family argument, which ultimately has to be an object of formal class “ebarraysFamily” with some specific slots that determine the behavior of the ‘family’.

For users who are content to use the predefined GG, LNN and LNNMV models, no further details than that given in the documentation for emfit are necessary. If you wish to create your own families, read on.

**Value**

Objects of class “ebarraysFamily” for the three predefined families Gamma-Gamma, Lognormal-Normal and Lognormal-Normal with modified variances.

**Objects from the Class**

Objects of class “ebarraysFamily” can be created by calls of the form `new("ebarraysFamily", ...)`. Predefined objects corresponding to the GG, LNN and LNNMV models can be created by `eb.createFamilyGG()`, `eb.createFamilyLNN()` and `eb.createFamilyLNNMV()`. The same effect is achieved by coercing from the strings “GG”, “LNN” and “LNNMV” by `as("GG", "ebarraysFamily"), as("LNN", "ebarraysFamily")` and `as("LNNMV", "ebarraysFamily")`.

**Slots**

An object of class “ebarraysFamily” extends the class "character" (representing a short hand name for the class) and should have the following slots (for more details see the source code):

- **description**: A not too long character string describing the family
- **link**: function that maps user-visible parameters to the parametrization that would be used in the optimization step (e.g. \( \log(\text{sigma}^2) \) for LNN). This allows the user to think in terms of familiar parametrization that may not necessarily be the best when optimizing w.r.t. those parameters.
- **invlink**: inverse of the link function
- **thetaInit**: function of a single argument data (matrix containing raw expression values), that calculates and returns as a numeric vector initial estimates of the parameters (in the parametrization used for optimization)
- **f0**: function taking arguments theta and a list called args. \( f_0 \) calculates the negative log likelihood at the given parameter value theta (again, in the parametrization used for optimization). This is called from emfit. When called, only genes with positive intensities across all samples are used.
- **f0.pp**: \( f_0.pp \) is essentially the same as \( f_0 \) except the terms common to the numerator and denominator when calculating posterior odds may be removed. It is called from postprob.
- **f0.arglist**: function that takes arguments data, patterns (of class “ebarraysPatterns”) and groupid (for LNNMV family only) and returns a list with two components, common.args and pattern.args. common.args is a list of arguments to \( f_0 \) that don’t change from one pattern to another, whereas pattern.args[[i]][[j]] is a similar list of arguments, but specific to the columns
in pattern[[i]][[j]]. Eventually, the two components will be combined for each pattern and used as the arg argument to f0.

logDensity: function of two arguments x (data vector, containing log expressions) and theta (parameters in user-visible parametrization). Returns log marginal density of the natural log of intensity for the corresponding theoretical model. Used in plotMarginal

lower.bound: vector of lower bounds for the argument theta of f0. Used in optim

upper.bound: vector of upper bounds for the argument theta of f0.

Author(s)
Ming Yuan, Ping Wang, Deepayan Sarkar, Michael Newton, and Christina Kendziorski

References


See Also
emfit, optim, plotMarginal

Examples
show(eb.createFamilyGG())
show(eb.createFamilyLNN())
show(eb.createFamilyLNNMV())

Description
Various plotting routines, used for diagnostic purposes
Usage

checkCCV(data, useRank = FALSE, f = 1/2)
checkModel(data, fit, model = c("gamma", "lognormal", "lnnmv"),
number = 9, nb = 10, cluster = 1, groupid = NULL)
checkVarsQQ(data, groupid, ...)
checkVarsMar(data, groupid, xlab, ylab, ...)
plotMarginal(fit, data, kernel = "rect", n = 100,
bw = "nrd0", adjust = 1, xlab, ylab,...)
plotCluster(fit, data, cond = NULL, ncolors = 123, sep=TRUE,
transform=NULL)

## S3 method for class 'ebarraysEMfit'
plot(x, data, plottype="cluster", ...)

Arguments

data data, as a “matrix” or “ExpressionSet”
useRank logical. If TRUE, ranks of means and c.v.-s are used in the scatterplot
f passed on to lowess
fit, x object of class “ebarraysEMfit”, typically produced by a call to emfit
model which theoretical model use for Q-Q plot. Partial string matching is allowed
number number of bins for checking model assumption.
nb number of data rows included in each bin for checking model assumption
cluster check model assumption for data in that cluster
groupid an integer vector indicating which group each sample belongs to. groupid for
samples not included in the analysis should be 0.
kernel, n, bw, adjust passed on to density
cond a vector specifying the condition for each replicate
ncolors different number of colors in the plot
xlab, ylab labels for x-axis and y-axis
sep whether or not to draw horizontal lines between clusters
transform a function to transform the original data in plotting
plottype a character string specifying the type of the plot. Available options are "cluster" and "marginal". The default plottype "cluster" employs function 'plotCluster' whereas the "marginal" plottype uses function 'plotMarginal'.

... extra arguments are passed to the qqmath, histogram and xyplot call used to produce the final result

Details

checkCCV checks the constant coefficient of variation assumption made in the GG and LNN models. checkModel generates QQ plots for subsets of (log) intensities in a small window. They are used to check the Log-Normal assumption on observation component of the LNN and LNNMV models and the Gamma assumption on observation component of the GG model. checkVarsQQ generates QQ plot for gene specific sample variances. It is used to check the assumption of a scaled inverse chi-square prior on gene specific variances, made in the LNNMV model. checkVarsMar is another
diagnostic tool to check this assumption. The density histogram of gene specific sample variances
and the density of the scaled inverse chi-square distribution with parameters estimated from data
will be plotted. checkMarginal generates predictive marginal distribution from fitted model and
compares with estimated marginal (kernel) density of data. Available for the GG and LNN models
only. plotCluster generate heatmap for gene expression data with clusters

Value

checkModel, checkVarsQQ and checkVarsMar return an object of class “trellis”, using function in
the Lattice package. Note that in certain situations, these may need to be explicitly 'print'-ed to
have any effect.

Author(s)

Ming Yuan, Ping Wang, Deepayan Sarkar, Michael Newton, and Christina Kendziorski

References

ability of expression ratios: Improving statistical inference about gene expression changes from
methods for comparing multiple groups using replicated gene expression profiles. Statistics in
Medicine 22:3899-3914.
Newton, M.A. and Kendziorski, C.M. Parametric Empirical Bayes Methods for Microarrays in The
analysis of gene expression data: methods and software. Eds. G. Parmigiani, E.S. Garrett, R.

See Also

emfit, lowess

emfit

Implements EM algorithm for gene expression mixture model

Description

Implements the EM algorithm for gene expression mixture model

Usage

emfit(data,
  family,
  hypotheses,
  ...)
**Arguments**

- **data**: a matrix
- **family**: an object of class "ebarraysFamily" or a character string which can be coerced to one. Currently, only the characters "GG" and "LNN", and "LNNMV" are valid. For LNNMV, a groupid is required. See below. Other families can be supplied by constructing them explicitly.
- **hypotheses**: an object of class "ebarraysPatterns" representing the hypotheses of interest. Such patterns can be generated by the function `ebPatterns`
- **cluster**: if `type=1`, `cluster` is a vector specifying the fixed cluster membership for each gene; if `type=2`, `cluster` specifies the number of clusters to be fitted
- **type**: if `type=1`, the cluster membership is fixed as input `cluster`; if `type=2`, fit the data with a fixed number of clusters
- **criterion**: only used when `type=2` and `cluster` contains more than one integers. All numbers of clusters provided in `cluster` will be fitted and the one that minimizes `criterion` will be returned. Possible values now are "BIC", "AIC" and "HQ"
- **cluster.init**: only used when `type=2`. Specify the initial clustering membership.
- **num.iter**: number of EM iterations
- **verbose**: logical or numeric (0,1,2) indicating desired level of information printed for the user
- **optim.control**: list passed unchanged to `optim` for finer control
- **groupid**: an integer vector indicating which group each sample belongs to, required in the "LNNMV" model. It does not depend on "hypotheses".

**Details**

There are many optional arguments. So a call might look more like this:

```r
efit(data, family, hypotheses, cluster, type=2, criterion="BIC", cluster.init = NULL, num.iter = 20, verbose = getOption("verbose"), optim.control = list(), ...)
```

**Value**

an object of class "ebarraysEMfit", that can be summarized by `show()` and used to generate posterior probabilities using `postprob`

**Author(s)**

Ming Yuan, Ping Wang, Deepayan Sarkar, Michael Newton, and Christina Kendziorski

**References**


See Also
ebPatterns, ebarraysFamily-class

Examples

```r
data(sample.ExpressionSet) ## from Biobase
eset <- exprs(sample.ExpressionSet)
patterns <- ebPatterns(c("1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1",
                           "1 1 1 1 1 1 1 1 1 1 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2")
                              
        )
        
        gg.fit <- emfit(data = eset, family = "GG", hypotheses = patterns, verbose = TRUE)
        show(gg.fit)
```

gould

A dataset of class matrix

Description

This dataset is part of a dataset from a study of gene expression patterns of mammary epithelial cells in a rat model of breast cancer, consisting of 5000 genes in 4 biological conditions; 10 arrays total.

Usage

data(gould)

Format

The data are originally from Affymetrix chips, subsequently processed by dChip and then exported to R for analysis.

Source

Dr. M.N. Gould’s laboratory in UW-Madison

Examples

data(gould)
postprob

Calculates posterior probabilities for expression patterns

Description

Takes the output from emfit and calculates the posterior probability of each of the hypotheses, for each gene.

Usage

postprob(fit, data, ...)

Arguments

- **fit**: output from emfit
- **data**: a numeric matrix or an object of class “ExpressionSet” containing the data, typically the same one used in the emfit fit supplied below.
- **...**: other arguments, ignored

Value

An object of class “ebarraysPostProb”. Slot joint is an three dimensional array of probabilities. Each element gives the posterior probability that a gene belongs to certain cluster and have certain pattern. cluster is a matrix of probabilities with number of rows given by the number of genes in data and as many columns as the number of clusters for the fit. pattern is a matrix of probabilities with number of rows given by the number of genes in data and as many columns as the number of patterns for the fit. It additionally contains a slot ‘hypotheses‘ containing these hypotheses.

Author(s)

Ming Yuan, Ping Wang, Deepayan Sarkar, Michael Newton, and Christina Kendziorski

References


See Also

emfit
Examples

```r
data(sample.ExpressionSet) ## from Biobase
eset <- exprs(sample.ExpressionSet)
patterns <- ebPatterns(c("1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 ",
"1 1 1 1 1 1 1 1 1 1 1 1 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2")

gg.fit <- emfit(data = eset, family = "GG", hypotheses = patterns, verbose = TRUE)
prob <- postprob(gg.fit,eset)
```

Description

Utility functions for the EBarrays package

Usage

```r
ebPatterns(x, ordered=FALSE)
```

Arguments

- `x`  
  x can be a character vector (of length > 2) (see example), or an arbitrary connection which should provide patterns, one line for each pattern. If `x` is a character vector of length 1, it is assumed to be the name of a file (since there’s no point in a patterns object with only one pattern) which is then opened and treated as a connection.

- `ordered`  
  logical variable specifying whether the pattern is ordered or not

Details

`ebPatterns` creates objects that represent a collection of hypotheses to be used by `emfit`.

Value

`ebPatterns` creates an Object of class “ebarraysPatterns”, to be used in other functions such as `emfit`. This is nothing more than a list (and can be treated as such as far as indexing goes) and is used only for method dispatch.

Author(s)

Ming Yuan, Ping Wang, Deepayan Sarkar, Michael Newton, and Christina Kendziorski

References


See Also

emfit

Examples

patterns <- ebPatterns(c("1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1","1 1 1 1 1 1 1 1 1 1 1 1 1 1 2 2 2 2 2 2 2 2 2 2 2 2"), TRUE)
show(patterns)
Index

*Topic **datasets**
  gould, 8

*Topic **models**
  ebarraysFamily-class, 2
  ebplots, 4
  emfit, 6
  postprob, 9
  utilities, 10
  checkCCV (ebplots), 4
  checkModel (ebplots), 4
  checkVarsMar (ebplots), 4
  checkVarsQQ (ebplots), 4
  coerce, character, ebarraysFamily-method (ebarraysFamily-class), 2
  crit.fun, 1
  density, 5
  eb.createFamilyGG (ebarraysFamily-class), 2
  eb.createFamilyLNN (ebarraysFamily-class), 2
  eb.createFamilyLMNMM (ebarraysFamily-class), 2
  ebarraysEMfit-class (emfit), 6
  ebarraysFamily-class, 2
  ebarraysPatterns-class (utilities), 10
  ebarraysPostProb-class (postprob), 9
  ebPatterns, 7, 8
  ebPatterns (utilities), 10
  ebplots, 4
  emfit, 3–6, 6, 9–11
  emfit, ExpressionSet, character, ebarraysPatterns-method (emfit), 6
  emfit, ExpressionSet, ebarraysFamily, ebarraysPatterns-method (emfit), 6
  emfit, matrix, character, ebarraysPatterns-method (emfit), 6
  emfit, matrix, ebarraysFamily, ebarraysPatterns-method (emfit), 6
  gould, 8
  lowess, 5, 6
  optim, 4, 7
  plot.ebarraysEMfit (ebplots), 4
  plotCluster (ebplots), 4
  plotMarginal, 4
  plotMarginal (ebplots), 4
  postprob, 7, 9
  postprob, ebarraysEMfit, ExpressionSet-method (postprob), 9
  postprob, ebarraysEMfit, matrix-method (postprob), 9
  show, ebarraysEMfit-method (emfit), 6
  show, ebarraysFamily-method (ebarraysFamily-class), 2
  show, ebarraysPatterns-method (utilities), 10
  show, ebarraysPostProb-method (postprob), 9
  utilities, 10