plgem

April 19, 2009

`LPSeset` *ExpressionSet for Testing PLGEM*

Description

This `ExpressionSet` object contains a subset of the microarray data used in References for **PLGEM** set up and validation. Briefly, it contains normalized gene expression values from a total of 6 hybridizations on MG-U74Av2 Affymetrix GeneChip microarrays. Two experimental conditions are represented in this dataset: The baseline condition (C) contains data of immature murine dendritic cells (4 replicates); the other condition (LPS) contains data of the same cells stimulated for 4 hours with LPS (2 replicates).

Usage

```
data(LPSeset)
```

Format

An object of class `ExpressionSet`.

Author(s)

Mattia Pelizzola ⟨mattia.pelizzola@gmail.com⟩

Norman Pavelka ⟨nxp@stowers-institute.org⟩

References

Pavelka N, Pelizzola M, Vizzardelli C, Capozzoli M, Splendiani A, Granucci F, Ricciardi-Castagnoli P. A power law global error model for the identification of differentially expressed genes in microarray data. BMC Bioinformatics. 2004 Dec 17;5:203.; http://www.biomedcentral.com/1471-2105/5/203

Examples

```
data(LPSeset)
head(exprs(LPSeset))
```

`plgem.deg` *Selection of differentially expressed genes/proteins using PLGEM*

Description

This function selects differentially expressed genes/proteins (DEG) at a given significance level 'delta', based on observed **PLGEM** signal-to-noise ratio (STN) values (typically obtained via a call to `plgem.obsStn`) and pre-computed p-values (typically obtained via a call to `plgem.pValue`).

Usage

```
plgem.deg(observedStn, plgemPval, delta=0.001, verbose=FALSE)
```

Arguments

`observedStn`	`matrix` of observed STN values; output of function `plgem.obsStn`.
`plgemPval`	`matrix` of p-values; output of function `plgem.pValue`.
`delta`	numeric vector; the significance level(s) to be used for the selection of DEG; value(s) must be between 0 and 1 (excluded).
`verbose`	`logical`; if `TRUE`, comments are printed out while running.

Details

This function allows for the selection of DEG by setting a significance cut-off on pre-calculated p-values. The significance level 'delta' roughly represents the false positive rate of the DEG selection, e.g. if a 'delta' of 0.001 is chosen in a microarray dataset with 10000 genes, on average 10 of the selected DEG are expected to be false positives.

Value

This function returns a list with a number of items equal to the number of different significance levels ('delta') used as input. Each item of this list is again a list, whose number of items correspond to the number of performed comparisons (i.e. the number of conditions in the starting `ExpressionSet` minus the baseline). Each of these second level list-items is a vector of observed STN values of the genes or proteins that passed the corresponding significance threshold in the corresponding comparison.

Author(s)

Mattia Pelizzola ⟨mattia.pelizzola@gmail.com⟩

Norman Pavelka ⟨nxp@stowers-institute.org⟩

References

Pavelka N, Pelizzola M, Vizzardelli C, Capozzoli M, Splendiani A, Granucci F, Ricciardi-Castagnoli P. A power law global error model for the identification of differentially expressed genes in microarray data. BMC Bioinformatics. 2004 Dec 17;5:203.; http://www.biomedcentral.com/1471-2105/5/203

Pavelka N, Fournier ML, Swanson SK, Pelizzola M, Ricciardi-Castagnoli P, Florens L, Washburn MP. Statistical similarities between transcriptomics and quantitative shotgun proteomics data. Mol Cell Proteomics. 2007 Nov 19; http://www.mcponline.org/cgi/content/abstract/M700240-MCP200v1

See Also

plgem.fit, plgem.obsStn, plgem.resampledStn, plgem.pValue, run.plgem

Examples

```
data(LPSeset)
LPSfit <- plgem.fit(data=LPSeset, fittingEval=TRUE)
LPSobsStn <- plgem.obsStn(data=LPSeset, plgemFit=LPSfit)
set.seed(123)
LPSresampledStn <- plgem.resampledStn(data=LPSeset, plgemFit=LPSfit)
LPSpValues <- plgem.pValue(LPSobsStn, LPSresampledStn)
LPSdegList <- plgem.deg(observedStn=LPSobsStn, plgemPval=LPSpValues, delta=0.001)
```

---

plgem.fit *PLGEM Fitting and Evaluation*

Description

Function for fitting and evaluating goodness of fit of **PLGEM** on a 'data' ExpressionSet, using the set of replicated samples identified by the 'fit.condition' condition of the 'covariateNumb' covariate. The range of gene expression values (or protein abundance levels) will be partitioned in 'p' intervals, and the model will be fit at the 'q'-th quantile of standard deviations in each partition.

Usage

```
plgem.fit(data, covariateNumb=1, fit.condition=1, p=10, q=0.5,
  fittingEval=FALSE, plot.file=FALSE, verbose=FALSE)
```

Arguments

data	an object of class ExpressionSet; see Details for important information on how the phenoData slot of this object will be interpreted by the function.
covariateNumb	integer (or coercible to integer); the covariate used to determine on which samples to fit the **PLGEM**.
fit.condition	integer (or coercible to integer); the condition used for **PLGEM** fitting, according to the order of unique values of the 'covariateNumb' covariate.
p	integer (or coercible to integer); number of intervals used to partition the expression value range.
q	numeric in [0,1]; the quantile of standard deviation used for **PLGEM** fitting.
fittingEval	logical; if TRUE, the fitting is evaluated generating a diagnostic plot.
plot.file	logical; if TRUE, a png file is written on the current working directory.
verbose	logical; if TRUE, comments are printed out while running.

Details

plgem.fit fits a Power Law Global Error Model (**PLGEM**) to an expression set and optionally evaluates the quality of the fit. This **PLGEM** aims to find the mathematical relationship between standard deviation and mean gene expression values (or protein abundance levels) in a set of replicated microarray (or proteomics) samples, according to the following power law:

ln(modeledSpread) = PLGEMslope * ln(mean) + PLGEMintercept

It has been demonstrated that this model fits to Affymetrix GeneChip datasets, as well as to datasets of normalized spectral counts obtained by mass spectrometry-based proteomics (see References for details).

The 'covariateNumb' covariate (the first one by default) of the phenoData of the ExpressionSet 'data' is expected to contain the necessary information about the experimental design. The values of this covariate must be sample labels, that have to be identical for samples to be treated as replicates.

plgem.fit returns 'SLOPE' and 'INTERCEPT' of the above described power law; moreover it returns the Pearson's correlation coefficient ('DATA.PEARSON') of ln(mean) vs. ln(sd) in the original data, as well as the adjusted R squared ('ADJ.R2.MP') of the **PLGEM** fitted to the modelling points.

If argument 'fittingEval' is TRUE, a graphical control of the goodness of the **PLGEM** fitting is produced and a plot containing four panels is generated. The top-left panel shows the power law, characterized by 'SLOPE' and 'INTERCEPT'. The top-right panel represents the distribution of model residuals. The bottom-left reports the contour plot of ranked residuals. The bottom-right panel finally shows the relationship between the distribution of observed residuals and the normal distribution. A good fit normally gives a horizontal symmetric rank-plot and a near normal distribution of residuals.

Value

plgem.fit returns a list of five numbers (see Details):

SLOPE	the slope of the fitted PLGEM.
INTERCEPT	the intercept of the fitted PLGEM.
DATA.PEARSON	the Pearson correlation coefficient of the linear model fitted on the original data.
ADJ.R2.MP	the adjusted R squared of PLGEM fitted on the modelling points.
FIT.CONDITION	the condition used for fitting PLGEM.

Author(s)

Mattia Pelizzola ⟨mattia.pelizzola@gmail.com⟩

Norman Pavelka ⟨nxp@stowers-institute.org⟩

References

Pavelka N, Pelizzola M, Vizzardelli C, Capozzoli M, Splendiani A, Granucci F, Ricciardi-Castagnoli P. A power law global error model for the identification of differentially expressed genes in microarray data. BMC Bioinformatics. 2004 Dec 17;5:203.; http://www.biomedcentral.com/1471-2105/5/203

Pavelka N, Fournier ML, Swanson SK, Pelizzola M, Ricciardi-Castagnoli P, Florens L, Washburn MP. Statistical similarities between transcriptomics and quantitative shotgun proteomics data. Mol Cell Proteomics. 2007 Nov 19; http://www.mcponline.org/cgi/content/abstract/M700240-MCP200v1

See Also

plgem.obsStn, plgem.resampledStn, plgem.pValue, plgem.deg, run.plgem

Examples

```
data(LPSeset)
LPSfit <- plgem.fit(data=LPSeset, fittingEval=FALSE)
sapply(LPSfit, function(x) return(as.vector(x)))
```

---

plgem.obsStn *Computation of Observed and Resampled PLGEM-STN statistics*

Description

These functions compute observed and resampled signal to noise ratio (STN) values using **PLGEM** fitting parameters (obtained via a call to function plgem.fit) to detect differential expression in an ExpressionSet 'data', containing either microarray or proteomics data.

Usage

```
plgem.obsStn(data, plgemFit, covariateNumb=1, baseline.condition=1,
  verbose=FALSE)
plgem.resampledStn(data, plgemFit, covariateNumb=1, baseline.condition=1,
  iterations="automatic", verbose=FALSE)
```

Arguments

data	an object of class ExpressionSet; see Details for important information on how the phenoData slot of this object will be interpreted by the function.
plgemFit	list; the output of 'plgem.fit'.
covariateNumb	integer (or coercible to integer); the covariate used to determine on which samples to fit the **PLGEM**.
baseline.condition	integer (or coercible to integer); the condition to be treated as the baseline.
verbose	logical; if TRUE, comments are printed out while running.
iterations	number of iterations for the resampling step; if "automatic" it is automatically determined.

Details

The 'covariateNumb' covariate (the 1st one by default) in the pData of the ExpressionSet 'data' is expected to contain the necessary information about the experimental design. The values of this covariate must be sample labels, that have to be identical for samples to be treated as replicates. In particular, the ExpressionSet 'data' must have at least two conditions in the 'covariateNumb' covariate; by default the first one is considered the baseline.

PLGEM-STN values are a measure of the degree of differential expression between a condition and the baseline:

PLGEM-STN = [mean(condition)-mean(baseline)] / [modeledSpread(condition)+modeledSpread(baseline)],

where: ln(modeledSpread) = PLGEMslope * ln(mean) + PLGEMintercept

`plgem.obsStn` determines the observed PLGEM-STN values for each gene or protein in 'data'. `plgem.resampledStn` determines the resampled **PLGEM** STN values for each gene or protein in 'data' using a resampling approach; see References for details. The number of iterations should be chosen depending on the number of replicates of the condition used for fitting the model.

Value

`plgem.obsStn` returns a `matrix` of observed **PLGEM** STN values. The `rownames` of this matrix are identical to the `rownames` of 'data'. The `colnames` represent the different experimental conditions that were compared to the baseline.

`plgem.resampledStn` returns a list with two items:

`RESAMPLED.STN` `matrix` of resampled PLGEM STN values, with `rownames` identical to those in 'data', and `colnames` representing the different number of replicates found in the different comparisons; see References for details.

`REPL.NUMBER` the number of replicates found for each experimental condition; see References for details.

Author(s)

Mattia Pelizzola ⟨mattia.pelizzola@gmail.com⟩

Norman Pavelka ⟨nxp@stowers-institute.org⟩

References

Pavelka N, Pelizzola M, Vizzardelli C, Capozzoli M, Splendiani A, Granucci F, Ricciardi-Castagnoli P. A power law global error model for the identification of differentially expressed genes in microarray data. BMC Bioinformatics. 2004 Dec 17;5:203.; http://www.biomedcentral.com/1471-2105/5/203

Pavelka N, Fournier ML, Swanson SK, Pelizzola M, Ricciardi-Castagnoli P, Florens L, Washburn MP. Statistical similarities between transcriptomics and quantitative shotgun proteomics data. Mol Cell Proteomics. 2007 Nov 19; http://www.mcponline.org/cgi/content/abstract/M700240-MCP200v1

See Also

`plgem.fit`, `plgem.pValue`, `plgem.deg`, `run.plgem`

Examples

```
data(LPSeset)
LPSfit <- plgem.fit(data=LPSeset)
LPSobsStn <- plgem.obsStn(data=LPSeset, plgemFit=LPSfit)
set.seed(123)
LPSresampledStn <- plgem.resampledStn(data=LPSeset, plgemFit=LPSfit)
plot(density(LPSresampledStn[["RESAMPLED.STN"]], bw=0.01), col="black", lwd=2,
  xlab="PLGEM STN values",
  main="Distribution of observed and resampled PLGEM STN values")
lines(density(LPSobsStn, bw=0.01), col="red")
legend("topright", legend=c("resampled", "observed"), col=c("black", "red"),
  lwd=2:1)
```

plgem.pValue *Computation of PLGEM p-values*

Description

This function computes p-values for observed PLGEM signal-to-noise ratio (STN) values (typically obtained via a call to plgem.obsStn) from resampled STN values (typically obtained via a call to plgem.resampledStn).

Usage

```
plgem.pValue(observedStn, plgemResampledStn, verbose=FALSE)
```

Arguments

observedStn	matrix of observed PLGEM STN values; output of plgem.obsStn.
plgemResampledStn	list; output of plgem.resampledStn.
verbose	logical; if TRUE, comments are printed out while running.

Details

The p-value of each given observed STN value is computed based on the quantile that the given value occupies in the corresponding distribution of resampled PLGEM STN values, based on the following relationship:

p-value = min(2*quantile, 2*(1-quantile))

Value

plgem.pValue returns a matrix with the same dimensions and dimnames as the input 'observedStn', where each entry represents the p-value of the corresponding observed PLGEM STN value.

Author(s)

Mattia Pelizzola ⟨mattia.pelizzola@gmail.com⟩

Norman Pavelka ⟨nxp@stowers-institute.org⟩

References

Pavelka N, Pelizzola M, Vizzardelli C, Capozzoli M, Splendiani A, Granucci F, Ricciardi-Castagnoli P. A power law global error model for the identification of differentially expressed genes in microarray data. BMC Bioinformatics. 2004 Dec 17;5:203; http://www.biomedcentral.com/1471-2105/5/203

Pavelka N, Fournier ML, Swanson SK, Pelizzola M, Ricciardi-Castagnoli P, Florens L, Washburn MP. Statistical similarities between transcriptomics and quantitative shotgun proteomics data. Mol Cell Proteomics. 2007 Nov 19; http://www.mcponline.org/cgi/content/abstract/M700240-MCP200v1

See Also

plgem.fit, plgem.obsStn, plgem.resampledStn, run.plgem

Examples

```
data(LPSeset)
LPSfit <- plgem.fit(data=LPSeset)
LPSobsStn <- plgem.obsStn(data=LPSeset, plgemFit=LPSfit)
head(LPSobsStn)
set.seed(123)
LPSresampledStn <- plgem.resampledStn(data=LPSeset, plgemFit=LPSfit)
LPSpValues <- plgem.pValue(LPSobsStn, LPSresampledStn)
head(LPSpValues)
```

---

plgem.write.summary

Write a list of differentially expressed genes/proteins to the disk

---

Description

This function writes the list of differentially expressed genes/proteins obtained via a call to either plgem.deg or run.plgem to a series of files in the current working directory.

Usage

```
plgem.write.summary(x, verbose=FALSE)
```

Arguments

x	list; the output of either plgem.deg or run.plgem, i.e. a list of list(s) of named vectors.
verbose	logical; if TRUE, comments are printed out while running.

Details

The gene or protein lists are written to the current working directory, using conveniently chosen filenames that reflect the specific comparisons that were performed (i.e. which experimental condition was compared to the baseline) and the specific significance threshold that were used in the DEG selection step.

Value

The function returns no value. It is called for its side effect to write files to the working directory.

Author(s)

Mattia Pelizzola ⟨mattia.pelizzola@gmail.com⟩ Norman Pavelka ⟨nxp@stowers-institute.org⟩

References

Pavelka N, Pelizzola M, Vizzardelli C, Capozzoli M, Splendiani A, Granucci F, Ricciardi-Castagnoli P. A power law global error model for the identification of differentially expressed genes in microarray data. BMC Bioinformatics. 2004 Dec 17;5:203.; http://www.biomedcentral.com/1471-2105/5/203

Pavelka N, Fournier ML, Swanson SK, Pelizzola M, Ricciardi-Castagnoli P, Florens L, Washburn MP. Statistical similarities between transcriptomics and quantitative shotgun proteomics data. Mol Cell Proteomics. 2007 Nov 19; http://www.mcponline.org/cgi/content/abstract/M700240-MCP200v1

See Also

plgem.deg, run.plgem

Examples

```
## Not run:
data(LPSeset)
LPSdegList <- run.plgem(LPSeset)
plgem.write.summary(LPSdegList, verbose=TRUE)
## End(Not run)
```

`run.plgem` *Wrapper for Power Law Global Error Model (PLGEM) analysis method*

Description

This function automatically performs **PLGEM** fitting and evaluation, determination of observed and resampled **PLGEM** STN values, and selection of differentially expressed genes/proteins (DEG) using the **PLGEM** method.

Usage

```
run.plgem(esdata, signLev=0.001, rank=100, covariateNumb=1,
  baselineCondition=1, Iterations="automatic", fitting.eval=TRUE,
  plotFile=FALSE, writeFiles=FALSE, Verbose=FALSE)
```

Arguments

`esdata`	an object of class `ExpressionSet`; see Details for important information on how the `phenoData` slot of this object will be interpreted by the function.
`signLev`	numeric vector; significance level(s) for the DEG selection. Value(s) must be in (0,1).
`rank`	`integer` (or coercible to `integer`); the number of genes or proteins to be selected according to their PLGEM-STN rank. Only used if number of available replicates is too small to perform resampling (see Details).
`covariateNumb`	`integer` (or coercible to `integer`); the covariate used to determine on which samples to fit plgem.

`baselineCondition`
: `integer` (or coercible to `integer`); the condition to be treated as the baseline.

`Iterations`
: number of iterations for the resampling step; if "automatic" it is automatically determined.

`fitting.eval`
: `logical`; if `TRUE`, the fitting is evaluated generating a diagnostic plot.

`plotFile`
: `logical`; if `TRUE`, the generated plot is written on a file.

`writeFiles`
: `logical`; if `TRUE`, the generated list of DEG is written on disk file(s).

`Verbose`
: `logical`; if `TRUE`, comments are printed out while running.

Details

The 'covariateNumb' covariate (the first one by default) of the `phenoData` of the `ExpressionSet` 'data' is expected to contain the necessary information about the experimental design. The values of this covariate must be sample labels, that have to be identical for samples to be treated as replicates. In particular, the `ExpressionSet` 'esdata' must have at least two conditions in the 'covariateNumb' covariate; by default the first one is considered the baseline.

The model is fitted on the most replicated condition. When more conditions exist with the max number of replicates, the condition providing the best fit is chosen.

If less than 3 replicates are provided for the condition used for fitting, then the selection is based on ranking according to the observed **PLGEM** STN values. In this case the first 'rank' genes or proteins are selected for each comparison.

Otherwise DEG are selected comparing the observed and resampled **PLGEM** STN values at the 'signLev' significance level(s), based on p-values obtained via a call to function `plgem.pValue`. See References for details.

Value

This function returns a list with a number of items that is equal to the number of different significance levels ('signLev') used as input. Each item is again a list, whose number of items correspond to the number of performed comparisons, i.e. the number of conditions defined in the `phenoData` of 'esdata' minus the baseline. In each list-item the values are the observed **PLGEM** STN values of the significantly changing genes or proteins, named according to the `rownames` of the `exprs` of 'esdata'.

Author(s)

Mattia Pelizzola ⟨mattia.pelizzola@gmail.com⟩

Norman Pavelka ⟨nxp@stowers-institute.org⟩

References

Pavelka N, Pelizzola M, Vizzardelli C, Capozzoli M, Splendiani A, Granucci F, Ricciardi-Castagnoli P. A power law global error model for the identification of differentially expressed genes in microarray data. BMC Bioinformatics. 2004 Dec 17;5:203.; http://www.biomedcentral.com/1471-2105/5/203

Pavelka N, Fournier ML, Swanson SK, Pelizzola M, Ricciardi-Castagnoli P, Florens L, Washburn MP. Statistical similarities between transcriptomics and quantitative shotgun proteomics data. Mol Cell Proteomics. 2007 Nov 19; http://www.mcponline.org/cgi/content/abstract/M700240-MCP200v1

See Also

plgem.fit, plgem.obsStn, plgem.resampledStn, plgem.pValue, plgem.write.summary

Examples

```
data(LPSeset)
set.seed(123)
LPSdegList <- run.plgem(esdata=LPSeset)
```

Index