

Package ‘PECA’

January 28, 2025

Type Package

Title Probe-level Expression Change Averaging

Version 1.42.0

Author Tomi Suomi, Jukka Hiissa, Laura L. Elo

Maintainer Tomi Suomi <tomi.suomi@utu.fi>

Imports ROTS, limma, affy, genefilter, preprocessCore,
aroma.affymetrix, aroma.core

Depends R (>= 3.3)

Suggests SpikeIn

Description Calculates Probe-level Expression Change Averages (PECA) to identify differential expression in Affymetrix gene expression microarray studies or in proteomic studies using peptide-level measurements respectively.

biocViews Software, Proteomics, Microarray, DifferentialExpression,
GeneExpression, ExonArray, DifferentialSplicing

License GPL (>= 2)

LazyLoad yes

git_url <https://git.bioconductor.org/packages/PECA>

git_branch RELEASE_3_20

git_last_commit 9fe2557

git_last_commit_date 2024-10-29

Repository Bioconductor 3.20

Date/Publication 2025-01-27

Contents

PECA	2
PECASI	4
Index	6

 PECA

PECA differential gene expression

Description

Calculates the PECA ordinary or modified t-statistic to determine differential expression between two groups of samples in Affymetrix gene expression studies or peptide-based proteomic studies.

Usage

```
## Read AffyBatch object
PECA_AffyBatch(affy=NULL, normalize=FALSE, log=TRUE, test="t", type="median",
  paired=FALSE, progress=FALSE)

## Read CEL-files
PECA_CEL(samplenames1=NULL, samplenames2=NULL, normalize=FALSE, log=TRUE, test="t",
  type="median", paired=FALSE, progress=FALSE)

## Read tab separated text file
PECA_tsv(file=NULL, samplenames1=NULL, samplenames2=NULL, normalize=FALSE, log=TRUE,
  test="t", type="median", paired=FALSE, progress=FALSE)

## Read dataframe
PECA_df(df=NULL, id=NULL, samplenames1=NULL, samplenames2=NULL, normalize=FALSE,
  log=TRUE, test="t", type="median", paired=FALSE, progress=FALSE)
```

Arguments

<code>affy</code>	AffyBatch object.
<code>normalize</code>	A character string indicating if ("quantile") or ("median") normalization is performed.
<code>log</code>	A logical indicating whether log2 scaling is performed.
<code>test</code>	A character string indicating whether the ordinary t-test ("t"), modified t-test ("modt"), or reproducibility-optimized test statistic ("rots") is performed.
<code>type</code>	A character string indicating whether ("median") or ("tukey") is used when calculating gene/protein values.
<code>paired</code>	A logical indicating whether a paired test is performed.
<code>file</code>	Filename of tab separated data.
<code>samplenames1</code>	A character vector containing the names of the .CEL-files/columns in the first group.
<code>samplenames2</code>	A character vector containing the names of the .CEL-files/columns in the second group.
<code>df</code>	Dataframe to be used as an input.
<code>id</code>	Column name of dataframe used for aggregating results.
<code>progress</code>	A logical indicating whether a progress bar is shown.

Details

PECA determines differential gene expression using directly the probe-level measurements from Affymetrix gene expression microarrays or proteomic datasets. An expression change between two groups of samples is first calculated for each probe/peptide on the array. The gene/protein-level expression changes are then defined as medians over the probe-level changes. For more details about the probe-level expression change averaging (PECA) procedure, see Elo et al. (2005), Laajala et al. (2009) and Suomi et al.

PECA calculates the probe-level expression changes using the ordinary or modified t-statistic. The ordinary t-statistic is calculated using the function `rowttests` in the Bioconductor `genefilter` package. The modified t-statistic is calculated using the linear modeling approach in the Bioconductor `limma` package. Both paired and unpaired tests are supported.

The significance of an expression change is determined based on the analytical p-value of the gene-level test statistic. Unadjusted p-values are reported along with the corresponding p-values looked up from beta distribution. The quality control and filtering of the data (e.g. based on low intensity or probe specificity) is left to the user.

Value

PECA returns a matrix which rows correspond to the genes under analysis and columns indicate the corresponding signal log-ratio (slr), t-statistic, p-value and FDR adjusted p-value.

References

- T. Suomi, G.L. Corthals, O. Nevalainen and L.L. Elo: Using peptide-level proteomics data for detecting differentially expressed proteins. 2015
- L.L. Elo, L. Lahti, H. Skottman, M. Kylaniemi, R. Lahesmaa and T. Aittokallio: Integrating probe-level expression changes across generations of Affymetrix arrays. *Nucleic Acids Research* 33(22), e193, 2005.
- E. Laajala, T. Aittokallio, R. Lahesmaa and L.L. Elo: Probe-level estimation improves the detection of differential splicing in Affymetrix exon array studies. *Genome Biology* 10(7), R77, 2009.
- H. Bengtsson, K. Simpson, J. Bullard and K. Hansen: *aroma.affymetrix*: A generic framework in R for analyzing small to very large Affymetrix data sets in bounded memory. Tech Report #745, Department of Statistics, University of California, Berkeley, 2008.

Examples

```
## Generate example data frame
df <- data.frame(id=c(rep("a",10),rep("b",10),rep("c",10)))
df$A1 <- rnorm(30, mean=50, sd=5)
df$A2 <- rnorm(30, mean=48, sd=5)
df$A3 <- rnorm(30, mean=50, sd=5)
df$B1 <- rnorm(30, mean=52, sd=5)
df$B2 <- rnorm(30, mean=53, sd=5)
df$B3 <- rnorm(30, mean=51, sd=5)

## Run the test
group1 <- c("A1","A2","A3")
group2 <- c("B1","B2","B3")
results <- PECA_df(df, group1, group2, id=id)
```

PECASI

*PECA splicing index***Description**

Calculates the PECA splicing index to determine differentially spliced exons between two groups of samples in Affymetrix exon array studies.

Usage

```
PECASI(path, dataFolder, chipType, cdfTag=NULL, samplenames1, samplenames2, test="t")
```

Arguments

path	A character string specifying the path of the working directory containing the expression and annotation data.
dataFolder	A character string specifying the name of the directory containing the raw expression data (.CEL-files).
chipType	A character string specifying the microarray (chip) type.
cdfTag	A character string indicating an optional suffix added to the name of the particular chip definition file (CDF).
samplenames1	A character vector containing the names of the .CEL-files in the first group without the extension .CEL.
samplenames2	A character vector containing the names of the .CEL-files in the second group without the extension .CEL. The paired samples are assumed to be in the same order in both of the vectors samplenames1 and samplenames2.
test	A character string indicating whether the ordinary ("t") or modified ("modt") t-test is performed.

Details

PECASI determines differential alternative splicing using directly the probe-level measurements from Affymetrix exon microarrays. Differential splicing between two groups of samples is first calculated for each probe on the array. The exon-level differential splicing is then defined as the median over the probe-level differences. For more details about the probe-level expression change averaging (PECA) procedure, see Elo et al. (2005), Elo et al. (2006) and Laajala et al.

The current implementation of PECASI calculates the probe-level differential splicing using the ordinary or modified t-statistic over splicing index values. The ordinary t-statistic is calculated using the function `rowttests` in the Bioconductor `genefilter` package. The modified t-statistic is calculated using the linear modeling approach in the Bioconductor `limma` package. The samples are assumed to be paired. For more details about the PECA splicing index procedure, see Laajala et al.

PECASI uses the `aroma.affymetrix` package to normalize and extract the probe-level data from the .CEL-files (Bengtsson et al. 2008). Therefore, it is important that the naming and structure of the data files follow exactly the rules specified in the `aroma.affymetrix` package.

The raw expression data (.CEL-files) need to be in the directory `rawData/<dataFolder>/<chipType>`, where `rawData` is a directory under the current working directory specified by the `path`, `dataFolder` is the name of the dataset given by the user, and `chipType` indicates the type of the microarray used in the experiment.

In addition to the expression data, a chip definition file (CDF) is required. The CDF-file(s) for a particular microarray type `chipType` need to be in the directory `annotationData/chipTypes/<chipType>`, where `annotationData` is a directory under the current working directory specified by the path. Besides the CDF-files provided by Affymetrix, various custom CDF-files are available for a particular microarray type. The different versions can be separated by adding a suffix `cdfTag` to the name of the CDF-file: `<chipType>,<cdfTag>.cdf`

The quality control and filtering of the data (e.g. based on low intensity or probe specificity) is left to the user.

Value

PECASI returns a matrix which rows correspond to the exons under analysis and columns indicate the corresponding splicing index (si), t-statistic, p-value and FDR adjusted p-value.

References

L.L. Elo, L. Lahti, H. Skottman, M. Kylaniemi, R. Lahesmaa and T. Aittokallio: Integrating probe-level expression changes across generations of Affymetrix arrays. *Nucleic Acids Research* 33(22), e193, 2005.

L.L. Elo, M. Katajamaa, R. Lund, M. Oresic, R. Lahesmaa and T. Aittokallio: Improving identification of differentially expressed genes by integrative analysis of Affymetrix and Illumina arrays. *OMICS A Journal of Integrative Biology* 10(3), 369–380, 2006.

E. Laajala, T. Aittokallio, R. Lahesmaa and L.L. Elo: Probe-level estimation improves the detection of differential splicing in Affymetrix exon array studies. *Genome Biology* 10(7), R77, 2009.

H. Bengtsson, K. Simpson, J. Bullard and K. Hansen: `aroma.affymetrix`: A generic framework in R for analyzing small to very large Affymetrix data sets in bounded memory. Tech Report #745, Department of Statistics, University of California, Berkeley, 2008.

See Also

[PECA](#)

Index

* **math**

PECA, [2](#)

PECASI, [4](#)

PECA, [2](#), [5](#)

PECA_AffyBatch (PECA), [2](#)

PECA_CEL (PECA), [2](#)

PECA_df (PECA), [2](#)

PECA_tsv (PECA), [2](#)

PECASI, [4](#)