DEGseq

April 20, 2011

`DEGexp2` *DEGexp2: Identifying Differentially Expressed Genes from gene expression data*

Description

This function is another (old) version of DEGexp. It takes the gene expression files as input instead of gene expression matrixs.

Usage

```
DEGexp2(geneExpFile1, geneCol1=1, expCol1=2, depth1=rep(0, length(expCol1)), gro
        geneExpFile2, geneCol2=1, expCol2=2, depth2=rep(0, length(expCol2)), gro
        header=TRUE, sep="", method=c("LRT", "CTR", "FET", "MARS", "MATR", "FC")
        pValue=1e-3, zScore=4, qValue=1e-3, foldChange=2,
        thresholdKind=1, outputDir="none", normalMethod=c("none", "loess", "medi
        replicate1="none", geneColR1=1, expColR1=2, depthR1=rep(0, length(expCol
        replicate2="none", geneColR2=1, expColR2=2, depthR2=rep(0, length(expCol
```

Arguments

`geneExpFile1`	file containing gene expression values for replicates of sample1 (or replicate1 when `method="CTR"`).
`geneCol1`	gene id column in geneExpFile1.
`expCol1`	expression value *columns* in geneExpFile1 for replicates of sample1 (numeric vector). *Note*: Each column corresponds to a replicate of sample1.
`depth1`	the total number of reads uniquely mapped to genome for each replicate of sample1 (numeric vector), default: take the total number of reads mapped to all annotated genes as the depth for each replicate.
`groupLabel1`	label of group1 on the plots.
`geneExpFile2`	file containing gene expression values for replicates of sample2 (or replicate2 when `method="CTR"`).
`geneCol2`	gene id column in geneExpFile2.

`expCol2` expression value *columns* in geneExpFile2 for replicates of sample2 (numeric vector).
Note: Each column corresponds to a replicate of sample2.

`depth2` the total number of reads uniquely mapped to genome for each replicate of sample2 (numeric vector),
default: take the total number of reads mapped to all annotated genes as the depth for each replicate.

`groupLabel2` label of group2 on the plots.

`header` a logical value indicating whether geneExpFile1 and geneExpFile2 contain the names of the variables as its first line. See `?read.table`.

`sep` the field separator character. If sep = "" (the default for read.table) the separator is *white space*, that is one or more spaces, tabs, newlines or carriage returns. See `?read.table`.

`method` method to identify differentially expressed genes. Possible methods are:

- `"LRT"`: Likelihood Ratio Test (Marioni et al. 2008),
- `"CTR"`: Check whether the variation between Technical Replicates can be explained by the random sampling model (Wang et al. 2009),
- `"FET"`: Fisher's Exact Test (Joshua et al. 2009),
- `"MARS"`: MA-plot-based method with Random Sampling model (Wang et al. 2009),
- `"MATR"`: MA-plot-based method with Technical Replicates (Wang et al. 2009),
- `"FC"` : Fold-Change threshold on MA-plot.

`pValue` pValue threshold (for the methods: `LRT, FET, MARS, MATR`).
only used when `thresholdKind=1`.

`zScore` zScore threshold (for the methods: `MARS, MATR`).
only used when `thresholdKind=2`.

`qValue` qValue threshold (for the methods: `LRT, FET, MARS, MATR`).
only used when `thresholdKind=3` or `thresholdKind=4`.

`thresholdKind`
the kind of threshold. Possible kinds are:

- `1`: pValue threshold,
- `2`: zScore threshold,
- `3`: qValue threshold (Benjamini et al. 1995),
- `4`: qValue threshold (Storey et al. 2003).
- `5`: qValue threshold (Storey et al. 2003)and Fold-Change threshold on MA-plot.

`foldChange` fold change threshold on MA-plot (for the method: `FC`).

`outputDir` the output directory.

`normalMethod` the normalization method: `"none", "loess", "median"` (Yang et al. 2002).
recommend: `"none"`.

`replicate1` file containing gene expression values for replicate batch1 (only used when `method="MATR"`).
Note: replicate1 and replicate2 are two (groups of) technical replicates of a sample.

`geneColR1` gene id column in the expression file for replicate batch1 (only used when `method="MATR"`).

`expColR1` expression value *columns* in the expression file for replicate batch1 (numeric vector) (only used when `method="MATR"`).

`depthR1` the total number of reads uniquely mapped to genome for each replicate in replicate batch1 (numeric vector),
default: take the total number of reads mapped to all annotated genes as the depth for each replicate (only used when `method="MATR"`).

`replicateLabel1` label of replicate batch1 on the plots (only used when `method="MATR"`).

`replicate2` file containing gene expression values for replicate batch2 (only used when `method="MATR"`).
Note: replicate1 and replicate2 are two (groups of) technical replicates of a sample.

`geneColR2` gene id column in the expression file for replicate batch2 (only used when `method="MATR"`).

`expColR2` expression value *columns* in the expression file for replicate batch2 (numeric vector) (only used when `method="MATR"`).

`depthR2` the total number of reads uniquely mapped to genome for each replicate in replicate batch2 (numeric vector),
default: take the total number of reads mapped to all annotated genes as the depth for each replicate (only used when `method="MATR"`).

`replicateLabel2` label of replicate batch2 on the plots (only used when `method="MATR"`).

`rawCount` a logical value indicating the gene expression values are based on raw read counts or normalized values.

References

Benjamini,Y. and Hochberg,Y (1995). Controlling the false discovery rate: a practical and powerful approach to multiple testing. *J. R. Stat. Soc. Ser. B* **57**, 289-300.

Jiang,H. and Wong,W.H. (2008) Statistical inferences for isoform expression in RNA-seq. *Bioinformatics*, **25**, 1026-1032.

Bloom,J.S. et al. (2009) Measuring differential gene expression by short read sequencing: quantitative comparison to 2-channel gene expression microarrays. *BMC Genomics*, **10**, 221.

Marioni,J.C. et al. (2008) RNA-seq: an assessment of technical reproducibility and comparison with gene expression arrays. *Genome Res.*, **18**, 1509-1517.

Storey,J.D. and Tibshirani,R. (2003) Statistical significance for genomewide studies. *Proc. Natl. Acad. Sci.* **100**, 9440-9445.

Wang,L.K. and et al. (2010) DEGseq: an R package for identifying differentially expressed genes from RNA-seq data, *Bioinformatics* **26**, 136 - 138.

Yang,Y.H. et al. (2002) Normalization for cDNA microarray data: a robust composite method addressing single and multiple slide systematic variation. *Nucleic Acids Research*, **30**, e15.

See Also

`DEGexp`, `DEGseq`, `getGeneExp`, `readGeneExp`, `GeneExpExample1000`, `GeneExpExample5000`.

Examples

```
## kidney: R1L1Kidney, R1L3Kidney, R1L7Kidney, R2L2Kidney, R2L6Kidney
## liver: R1L2Liver, R1L4Liver, R1L6Liver, R1L8Liver, R2L3Liver

geneExpFile <- system.file("extdata", "GeneExpExample5000.txt", package="DEGseq")
outputDir <- file.path(tempdir(), "DEGexpExample")
exp <- readGeneExp(file=geneExpFile, geneCol=1, valCol=c(7,9,12,15,18))
exp[30:35,]
exp <- readGeneExp(file=geneExpFile, geneCol=1, valCol=c(8,10,11,13,16))
exp[30:35,]
DEGexp2(geneExpFile1=geneExpFile, geneCol1=1, expCol1=c(7,9,12,15,18), groupLabel1="kid
        geneExpFile2=geneExpFile, geneCol2=1, expCol2=c(8,10,11,13,16), groupLabel2="li
        method="MARS", outputDir=outputDir)
cat("outputDir:", outputDir, "\n")
```

`DEGexp` *DEGexp: Identifying Differentially Expressed Genes from gene expression data*

Description

This function is used to identify differentially expressed genes when users already have the gene expression values (or the number of reads mapped to each gene).

Usage

```
DEGexp(geneExpMatrix1, geneCol1=1, expCol1=2, depth1=rep(0, length(expCol1)), gr
       geneExpMatrix2, geneCol2=1, expCol2=2, depth2=rep(0, length(expCol2)), gr
       method=c("LRT", "CTR", "FET", "MARS", "MATR", "FC"),
       pValue=1e-3, zScore=4, qValue=1e-3, foldChange=2,
       thresholdKind=1, outputDir="none", normalMethod=c("none", "loess", "media
       replicateExpMatrix1=NULL, geneColR1=1, expColR1=2, depthR1=rep(0, length(
       replicateExpMatrix2=NULL, geneColR2=1, expColR2=2, depthR2=rep(0, length(
```

Arguments

`geneExpMatrix1`
: gene expression matrix for replicates of sample1 (or replicate1 when `method="CTR"`).

`geneCol1`
: gene id column in geneExpMatrix1.

`expCol1`
: expression value *columns* in geneExpMatrix1 for replicates of sample1 (numeric vector).
Note: Each column corresponds to a replicate of sample1.

`depth1`
: the total number of reads uniquely mapped to genome for each replicate of sample1 (numeric vector),
default: take the total number of reads mapped to all annotated genes as the depth for each replicate.

`groupLabel1`
: label of group1 on the plots.

`geneExpMatrix2`
: gene expression matrix for replicates of sample2 (or replicate2 when `method="CTR"`).

`geneCol2` gene id column in geneExpMatrix2.

`expCol2` expression value *columns* in geneExpMatrix2 for replicates of sample2 (numeric vector).
Note: Each column corresponds to a replicate of sample2.

`depth2` the total number of reads uniquely mapped to genome for each replicate of sample2 (numeric vector),
default: take the total number of reads mapped to all annotated genes as the depth for each replicate.

`groupLabel2` label of group2 on the plots.

`method` method to identify differentially expressed genes. Possible methods are:

- `"LRT"`: Likelihood Ratio Test (Marioni et al. 2008),
- `"CTR"`: Check whether the variation between Technical Replicates can be explained by the random sampling model (Wang et al. 2009),
- `"FET"`: Fisher's Exact Test (Joshua et al. 2009),
- `"MARS"`: MA-plot-based method with Random Sampling model (Wang et al. 2009),
- `"MATR"`: MA-plot-based method with Technical Replicates (Wang et al. 2009),
- `"FC"` : Fold-Change threshold on MA-plot.

`pValue` pValue threshold (for the methods: `LRT, FET, MARS, MATR`).
only used when `thresholdKind=1`.

`zScore` zScore threshold (for the methods: `MARS, MATR`).
only used when `thresholdKind=2`.

`qValue` qValue threshold (for the methods: `LRT, FET, MARS, MATR`).
only used when `thresholdKind=3` or `thresholdKind=4`.

`thresholdKind`
the kind of threshold. Possible kinds are:

- `1`: pValue threshold,
- `2`: zScore threshold,
- `3`: qValue threshold (Benjamini et al. 1995),
- `4`: qValue threshold (Storey et al. 2003).
- `5`: qValue threshold (Storey et al. 2003)and Fold-Change threshold on MA-plot.

`foldChange` fold change threshold on MA-plot (for the method: `FC`).

`outputDir` the output directory.

`normalMethod` the normalization method: `"none", "loess", "median"` (Yang et al. 2002).
recommend: `"none"`.

`replicateExpMatrix1`
matrix containing gene expression values for replicate batch1 (only used when `method="MATR"`).
Note: replicate1 and replicate2 are two (groups of) technical replicates of a sample.

`geneColR1` gene id column in the expression matrix for replicate batch1 (only used when `method="MATR"`).

`expColR1` expression value *columns* in the expression matrix for replicate batch1 (numeric vector) (only used when `method="MATR"`).

depthR1 the total number of reads uniquely mapped to genome for each replicate in replicate batch1 (numeric vector),
default: take the total number of reads mapped to all annotated genes as the depth for each replicate (only used when `method="MATR"`).

replicateLabel1
label of replicate batch1 on the plots (only used when `method="MATR"`).

replicateExpMatrix2
matrix containing gene expression values for replicate batch2 (only used when `method="MATR"`).
Note: replicate1 and replicate2 are two (groups of) technical replicates of a sample.

geneColR2 gene id column in the expression matrix for replicate batch2 (only used when `method="MATR"`).

expColR2 expression value *columns* in the expression matrix for replicate batch2 (numeric vector) (only used when `method="MATR"`).

depthR2 the total number of reads uniquely mapped to genome for each replicate in replicate batch2 (numeric vector),
default: take the total number of reads mapped to all annotated genes as the depth for each replicate (only used when `method="MATR"`).

replicateLabel2
label of replicate batch2 on the plots (only used when `method="MATR"`).

rawCount a logical value indicating the gene expression values are based on raw read counts or normalized values.

References

Benjamini,Y. and Hochberg,Y (1995). Controlling the false discovery rate: a practical and powerful approach to multiple testing. *J. R. Stat. Soc. Ser. B* **57**, 289-300.

Jiang,H. and Wong,W.H. (2008) Statistical inferences for isoform expression in RNA-seq. *Bioinformatics*, **25**, 1026-1032.

Bloom,J.S. et al. (2009) Measuring differential gene expression by short read sequencing: quantitative comparison to 2-channel gene expression microarrays. *BMC Genomics*, **10**, 221.

Marioni,J.C. et al. (2008) RNA-seq: an assessment of technical reproducibility and comparison with gene expression arrays. *Genome Res.*, **18**, 1509-1517.

Storey,J.D. and Tibshirani,R. (2003) Statistical significance for genomewide studies. *Proc. Natl. Acad. Sci.* **100**, 9440-9445.

Wang,L.K. and et al. (2010) DEGseq: an R package for identifying differentially expressed genes from RNA-seq data, *Bioinformatics* **26**, 136 - 138.

Yang,Y.H. et al. (2002) Normalization for cDNA microarray data: a robust composite method addressing single and multiple slide systematic variation. *Nucleic Acids Research*, **30**, e15.

See Also

`DEGexp2`, `DEGseq`, `getGeneExp`, `readGeneExp`, `GeneExpExample1000`, `GeneExpExample5000`.

Examples

```
## kidney: R1L1Kidney, R1L3Kidney, R1L7Kidney, R2L2Kidney, R2L6Kidney
## liver: R1L2Liver, R1L4Liver, R1L6Liver, R1L8Liver, R2L3Liver
```

```
geneExpFile <- system.file("extdata", "GeneExpExample5000.txt", package="DEGseq")
cat("geneExpFile:", geneExpFile, "\n")
outputDir <- file.path(tempdir(), "DEGexpExample")
geneExpMatrix1 <- readGeneExp(file=geneExpFile, geneCol=1, valCol=c(7,9,12,15,18))
geneExpMatrix2 <- readGeneExp(file=geneExpFile, geneCol=1, valCol=c(8,10,11,13,16))
geneExpMatrix1[30:32,]
geneExpMatrix2[30:32,]
DEGexp(geneExpMatrix1=geneExpMatrix1, geneCol1=1, expCol1=c(2,3,4,5,6), groupLabel1="ki
       geneExpMatrix2=geneExpMatrix2, geneCol2=1, expCol2=c(2,3,4,5,6), groupLabel2="li
       method="LRT", outputDir=outputDir)
cat("outputDir:", outputDir, "\n")
```

`DEGseq.aln` *DEGseq.aln: Identify Differentially Expressed Genes from RNA-seq data (deal with the objects of class AlignedRead)*

Description

This function is used to identify differentially expressed genes from RNA-seq data. It takes uniquely mapped reads from RNA-seq data for the two samples with a gene annotation as input. So users should map the reads (obtained from sequencing libraries of the samples) to the corresponding genome in advance.

Usage

```
DEGseq.aln(alnBatch1, alnBatch2,
           strandInfo=FALSE, refFlat, groupLabel1="group1", groupLabel2="group2"
           method=c("LRT", "CTR", "FET", "MARS", "MATR", "FC"),
           pValue=1e-3, zScore=4, qValue=1e-3, foldChange=2, thresholdKind=1,
           outputDir="none", normalMethod=c("none", "loess", "median"),
           depthKind=1, replicateAlnBatch1=NULL, replicateAlnBatch2=NULL,
           replicateLabel1="replicate1", replicateLabel2="replicate2")
```

Arguments

`alnBatch1`	list containing uniquely mapping reads (objects of class AlignedRead) for technical replicates of sample1 (or replicate1 when `method="CTR"`).
`alnBatch2`	list containing uniquely mapping reads (objects of class AlignedRead) for technical replicates of sample2 (or replicate2 when `method="CTR"`).
`strandInfo`	whether the strand information was retained during the cloning of the cDNAs. • `"TRUE"` : retained, • `"FALSE"`: not retained.
`refFlat`	gene annotation file in UCSC refFlat format. See http://genome.ucsc.edu/goldenPath/gbdDescriptionsOld.html#RefFlat.
`groupLabel1`	label of group1 on the plots.
`groupLabel2`	label of group2 on the plots.
`method`	method to identify differentially expressed genes. Possible methods are: • `"LRT"`: Likelihood Ratio Test (Marioni et al. 2008),

- "CTR": Check whether the variation between two Technical Replicates can be explained by the random sampling model (Wang et al. 2009),
- "FET": Fisher's Exact Test (Joshua et al. 2009),
- "MARS": MA-plot-based method with Random Sampling model (Wang et al. 2009),
- "MATR": MA-plot-based method with Technical Replicates (Wang et al. 2009),
- "FC" : Fold-Change threshold on MA-plot.

pValue — pValue threshold (for the methods: LRT, FET, MARS, MATR).
only used when thresholdKind=1.

zScore — zScore threshold (for the methods: MARS, MATR).
only used when thresholdKind=2.

qValue — qValue threshold (for the methods: LRT, FET, MARS, MATR).
only used when thresholdKind=3 or thresholdKind=4.

thresholdKind — the kind of threshold. Possible kinds are:

- 1: pValue threshold,
- 2: zScore threshold,
- 3: qValue threshold (Benjamini et al. 1995),
- 4: qValue threshold (Storey et al. 2003).

foldChange — fold change threshold on MA-plot (for the method: FC).

outputDir — the output directory.

normalMethod — the normalization method: "none", "loess", "median" (Yang,Y.H. et al. 2002).
recommend: "none".

depthKind — 1: take the total number of reads uniquely mapped to genome as the depth for each replicate,
0: take the total number of reads uniquely mapped to all annotated genes as the depth for each replicate.
We recommend taking depthKind=1, especially when the genes in annotation file are part of all genes.

replicateAlnBatch1 — list containing uniquely mapping reads (objects of class AlignedRead) obtained from replicate batch1 (only used when method="MATR").

replicateAlnBatch2 — list containing uniquely mapping reads (objects of class AlignedRead) obtained from replicate batch2 (only used when method="MATR").

replicateLabel1 — label of replicate batch1 on the plots (only used when method="MATR").

replicateLabel2 — label of replicate batch2 on the plots (only used when method="MATR").

Note

Users should use DEGseq instead of this function when the short reads are too many to be loaded into memory.

References

Benjamini,Y. and Hochberg,Y. (1995) Controlling the false discovery rate: a practical and powerful approach to multiple testing. *J. R. Stat. Soc. Ser. B* **57**, 289-300.

Jiang,H. and Wong,W.H. (2009) Statistical inferences for isoform expression in RNA-seq. *Bioinformatics*, **25**, 1026-1032.

Bloom,J.S. et al. (2009) Measuring differential gene expression by short read sequencing: quantitative comparison to 2-channel gene expression microarrays. *BMC Genomics*, **10**, 221.

Marioni,J.C. et al. (2008) RNA-seq: an assessment of technical reproducibility and comparison with gene expression arrays. *Genome Res.*, **18**, 1509-1517.

Storey,J.D. and Tibshirani,R. (2003) Statistical significance for genomewide studies. *Proc. Natl. Acad. Sci.* **100**, 9440-9445.

Wang,L.K. and et al. (2010) DEGseq: an R package for identifying differentially expressed genes from RNA-seq data, *Bioinformatics* **26**, 136 - 138.

Yang,Y.H. et al. (2002) Normalization for cDNA microarray data: a robust composite method addressing single and multiple slide systematic variation. *Nucleic Acids Research*, **30**, e15.

See Also

DEGexp, DEGseq, getGeneExp.aln, readGeneExp, kidneyChr21.bed, liverChr21.bed, refFlatChr21.

Examples

```
kidneyR1L1 <- system.file("extdata", "kidneyChr21Bowtie.txt", package="DEGseq")
liverR1L2  <- system.file("extdata", "liverChr21Bowtie.txt", package="DEGseq")
refFlat    <- system.file("extdata", "refFlatChr21.txt", package="DEGseq")
kidneyR1L1_aln <- ShortRead::readAligned(dirname(kidneyR1L1), basename(kidneyR1L1), typ
liverR1L2_aln <- ShortRead::readAligned(dirname(liverR1L2), basename(liverR1L2), type="
alnBatch1 <- list(kidneyR1L1_aln)  ## only use the data from kidneyR1L1 and liverR1L2
alnBatch2 <- list(liverR1L2_aln)
outputDir <- file.path(tempdir(), "DEGseqAlnExample")
DEGseq.aln(alnBatch1, alnBatch2, refFlat=refFlat, outputDir=outputDir, method="MARS")
cat("outputDir:", outputDir, "\n")
```

DEGseq *DEGseq: Identify Differentially Expressed Genes from RNA-seq data*

Description

This function is used to identify differentially expressed genes from RNA-seq data. It takes uniquely mapped reads from RNA-seq data for the two samples with a gene annotation as input. So users should map the reads (obtained from sequencing libraries of the samples) to the corresponding genome in advance.

Usage

```
DEGseq(mapResultBatch1, mapResultBatch2, fileFormat="bed", readLength=32,
       strandInfo=FALSE, refFlat, groupLabel1="group1", groupLabel2="group2",
       method=c("LRT", "CTR", "FET", "MARS", "MATR", "FC"),
       pValue=1e-3, zScore=4, qValue=1e-3, foldChange=2, thresholdKind=1,
       outputDir="none", normalMethod=c("none", "loess", "median"),
       depthKind=1, replicate1="none", replicate2="none",
       replicateLabel1="replicate1", replicateLabel2="replicate2")
```

Arguments

`mapResultBatch1`
: vector containing uniquely mapping result files for technical replicates of sample1 (or replicate1 when `method="CTR"`).

`mapResultBatch2`
: vector containing uniquely mapping result files for technical replicates of sample2 (or replicate2 when `method="CTR"`).

`fileFormat`
: file format: `"bed"` or `"eland"`.
example of `"bed"` format: `chr12 7 38 readID 2 +`
example of `"eland"` format: `readID chr12.fa 7 U2 F`
Note: The field separator character is `TAB`. And the files must follow the format as one of the examples.

`readLength`
: the length of the reads (only used if `fileFormat="eland"`).

`strandInfo`
: whether the strand information was retained during the cloning of the cDNAs.
  - `"TRUE"` : retained,
  - `"FALSE"`: not retained.

`refFlat`
: gene annotation file in UCSC refFlat format.
See http://genome.ucsc.edu/goldenPath/gbdDescriptionsOld.html#RefFlat.

`groupLabel1`
: label of group1 on the plots.

`groupLabel2`
: label of group2 on the plots.

`method`
: method to identify differentially expressed genes. Possible methods are:
  - `"LRT"`: Likelihood Ratio Test (Marioni et al. 2008),
  - `"CTR"`: Check whether the variation between two Technical Replicates can be explained by the random sampling model (Wang et al. 2009),
  - `"FET"`: Fisher's Exact Test (Joshua et al. 2009),
  - `"MARS"`: MA-plot-based method with Random Sampling model (Wang et al. 2009),
  - `"MATR"`: MA-plot-based method with Technical Replicates (Wang et al. 2009),
  - `"FC"` : Fold-Change threshold on MA-plot.

`pValue`
: pValue threshold (for the methods: `LRT, FET, MARS, MATR`).
only used when `thresholdKind=1`.

`zScore`
: zScore threshold (for the methods: `MARS, MATR`).
only used when `thresholdKind=2`.

`qValue`
: qValue threshold (for the methods: `LRT, FET, MARS, MATR`).
only used when `thresholdKind=3` or `thresholdKind=4`.

`thresholdKind`
: the kind of threshold. Possible kinds are:
  - `1`: pValue threshold,
  - `2`: zScore threshold,
  - `3`: qValue threshold (Benjamini et al. 1995),
  - `4`: qValue threshold (Storey et al. 2003).

`foldChange`
: fold change threshold on MA-plot (for the method: `FC`).

`outputDir`
: the output directory.

`normalMethod`
: the normalization method: `"none", "loess", "median"` (Yang,Y.H. et al. 2002).
recommend: `"none"`.

`depthKind`
: `1`: take the total number of reads uniquely mapped to genome as the depth for each replicate,
`0`: take the total number of reads uniquely mapped to all annotated genes as the depth for each replicate.
We recommend taking `depthKind=1`, especially when the genes in annotation file are part of all genes.

`replicate1`
: files containing uniquely mapped reads obtained from replicate batch1 (only used when `method="MATR"`).

`replicate2`
: files containing uniquely mapped reads obtained from replicate batch2 (only used when `method="MATR"`).

`replicateLabel1`
: label of replicate batch1 on the plots (only used when `method="MATR"`).

`replicateLabel2`
: label of replicate batch2 on the plots (only used when `method="MATR"`).

References

Benjamini,Y. and Hochberg,Y. (1995) Controlling the false discovery rate: a practical and powerful approach to multiple testing. *J. R. Stat. Soc. Ser. B* **57**, 289-300.

Jiang,H. and Wong,W.H. (2009) Statistical inferences for isoform expression in RNA-seq. *Bioinformatics*, **25**, 1026-1032.

Bloom,J.S. et al. (2009) Measuring differential gene expression by short read sequencing: quantitative comparison to 2-channel gene expression microarrays. *BMC Genomics*, **10**, 221.

Marioni,J.C. et al. (2008) RNA-seq: an assessment of technical reproducibility and comparison with gene expression arrays. *Genome Res.*, **18**, 1509-1517.

Storey,J.D. and Tibshirani,R. (2003) Statistical significance for genomewide studies. *Proc. Natl. Acad. Sci.* **100**, 9440-9445.

Wang,L.K. and et al. (2010) DEGseq: an R package for identifying differentially expressed genes from RNA-seq data, *Bioinformatics* **26**, 136 - 138.

Yang,Y.H. et al. (2002) Normalization for cDNA microarray data: a robust composite method addressing single and multiple slide systematic variation. *Nucleic Acids Research*, **30**, e15.

See Also

`DEGexp`, `DEGseq.aln`, `getGeneExp`, `readGeneExp`, `kidneyChr21.bed`, `liverChr21.bed`, `refFlatChr21`.

Examples

```
kidneyR1L1 <- system.file("extdata", "kidneyChr21.bed.txt", package="DEGseq")
liverR1L2  <- system.file("extdata", "liverChr21.bed.txt", package="DEGseq")
refFlat    <- system.file("extdata", "refFlatChr21.txt", package="DEGseq")
mapResultBatch1 <- c(kidneyR1L1)  ## only use the data from kidneyR1L1 and liverR1L2
mapResultBatch2 <- c(liverR1L2)
outputDir <- file.path(tempdir(), "DEGseqExample")
DEGseq(mapResultBatch1, mapResultBatch2, fileFormat="bed", refFlat=refFlat,
       outputDir=outputDir, method="LRT")
cat("outputDir:", outputDir, "\n")
```

GeneExpExample1000 *GeneExpExample1000*

Description

GeneExpExample1000.txt includes the first 1000 lines in SupplementaryTable2.txt which is a supplementary file for Marioni,J.C. et al. (2008) (http://genome.cshlp.org/content/18/9/1509/suppl/DC1).

References

Marioni,J.C. et al. (2008) RNA-seq: an assessment of technical reproducibility and comparison with gene expression arrays. *Genome Res.*, **18**, 1509-1517.

See Also

DEGexp, getGeneExp, readGeneExp, samWrapper, GeneExpExample5000.

GeneExpExample5000 *GeneExpExample5000*

Description

GeneExpExample5000.txt includes the first 5000 lines in SupplementaryTable2.txt which is a supplementary file for Marioni,J.C. et al. (2008) (http://genome.cshlp.org/content/18/9/1509/suppl/DC1).

References

Marioni,J.C. et al. (2008) RNA-seq: an assessment of technical reproducibility and comparison with gene expression arrays. *Genome Res.*, **18**, 1509-1517.

See Also

DEGexp, getGeneExp, readGeneExp, samWrapper, GeneExpExample1000.

`getGeneExp.aln` *getGeneExp.aln: Count the number of reads and calculate the RPKM for each gene (deal with the objects of class AlignedRead)*

Description

This function is used to count the number of reads and calculate the RPKM for each gene. It takes uniquely mapped reads from RNA-seq data for a sample with a gene annotation file as input. So users should map the reads (obtained from sequencing library of the sample) to the corresponding genome in advance.

Usage

```
getGeneExp.aln(alnBatch, strandInfo=FALSE,
               refFlat, output=paste(mapResultBatch[1],".exp",sep=""), min.overl
```

Arguments

`alnBatch`	list containing the objects of class AlignedRead (package:ShortRead).
`strandInfo`	whether the strand information was retained during the cloning of the cDNAs. • `"TRUE"` : retained, • `"FALSE"`: not retained.
`refFlat`	gene annotation file in UCSC refFlat format. See `http://genome.ucsc.edu/goldenPath/gbdDescriptionsOld.html#RefFlat`.
`output`	the output file.
`min.overlapPercent`	the minimum percentage of the overlapping length for a read and an exon over the length of the read itself, for counting this read from the exon. should be <=1. `0`: at least `1` bp overlap between a read and an exon.

Note

Users should use `getGeneExp` instead of this function when the short reads are too many to be loaded into memory. This function sums up the numbers of reads coming from all exons of a specific gene (according to the known gene annotation) as the gene expression value. The exons may include the 5'-UTR, protein coding region, and 3'-UTR of a gene. All introns are ignored for a gene for the sequenced reads are from the spliced transcript library. If a read falls in an exon (usually, a read is shorter than an exon), the read count for this exon plus 1. If a read is crossing the boundary of an exon, users can tune the parameter `min.overlapPercent`, which is the minimum percentage of the overlapping length for a read and an exon over the length of the read itself, for counting this read from the exon. The method use the union of all possible exons for calculating the length for each gene.

References

Mortazavi,A. et al. (2008) Mapping and quantifying mammalian transcriptomes by RNA-seq. *Nat. Methods*, **5**, 621-628.

See Also

DEGexp, DEGseq, getGeneExp, readGeneExp, kidneyChr21.bed, liverChr21.bed, refFlatChr21.

Examples

```
library(ShortRead)
kidneyR1L1 <- system.file("extdata", "kidneyChr21Bowtie.txt", package="DEGseq")
refFlat    <- system.file("extdata", "refFlatChr21.txt", package="DEGseq")
kidneyR1L1_aln <- ShortRead::readAligned(dirname(kidneyR1L1), basename(kidneyR1L1), typ
alnBatch <- list(kidneyR1L1_aln)
output <- file.path(tempdir(), "kidneyChr21Bowtie.exp")
exp <- getGeneExp.aln(alnBatch, refFlat=refFlat, output=output)
write.table(exp[30:35,], row.names=FALSE)
cat("output: ", output, "\n")
```

`getGeneExp` *getGeneExp: Count the number of reads and calculate the RPKM for each gene*

Description

This function is used to count the number of reads and calculate the RPKM for each gene. It takes uniquely mapped reads from RNA-seq data for a sample with a gene annotation file as input. So users should map the reads (obtained from sequencing library of the sample) to the corresponding genome in advance.

Usage

```
getGeneExp(mapResultBatch, fileFormat="bed", readLength=32, strandInfo=FALSE,
           refFlat, output=paste(mapResultBatch[1],".exp",sep=""), min.overlapPe
```

Arguments

`mapResultBatch`
: vector containing uniquely mapping result files for a sample.
Note: The sample can have multiple technical replicates.

`fileFormat`
: file format: `"bed"` or `"eland"`.
example of `"bed"` format: `chr12 7 38 readID 2 +`
example of `"eland"` format: `readID chr12.fa 7 U2 F`
Note: The field separator character is `TAB`. And the files must follow the format as one of the examples.

`readLength`
: the length of the reads (only used if `fileFormat="eland"`).

`strandInfo`
: whether the strand information was retained during the cloning of the cDNAs.
  - `"TRUE"` : retained,
  - `"FALSE"`: not retained.

`refFlat`
: gene annotation file in UCSC refFlat format.
See http://genome.ucsc.edu/goldenPath/gbdDescriptionsOld.html#RefFlat.

`output`
: the output file.

`min.overlapPercent`
: the minimum percentage of the overlapping length for a read and an exon over the length of the read itself, for counting this read from the exon. should be <=1. `0`: at least `1` bp overlap between a read and an exon.

Note

This function sums up the numbers of reads coming from all exons of a specific gene (according to the known gene annotation) as the gene expression value. The exons may include the 5'-UTR, protein coding region, and 3'-UTR of a gene. All introns are ignored for a gene for the sequenced reads are from the spliced transcript library. If a read falls in an exon (usually, a read is shorter than an exon), the read count for this exon plus 1. If a read is crossing the boundary of an exon, users can tune the parameter `min.overlapPercent`, which is the minimum percentage of the overlapping length for a read and an exon over the length of the read itself, for counting this read from the exon. The method use the union of all possible exons for calculating the length for each gene.

References

Mortazavi,A. et al. (2008) Mapping and quantifying mammalian transcriptomes by RNA-seq. *Nat. Methods*, **5**, 621-628.

See Also

`DEGexp`, `DEGseq`, `getGeneExp.aln`, `readGeneExp`, `kidneyChr21.bed`, `liverChr21.bed`, `refFlatChr21`.

Examples

```
kidneyR1L1 <- system.file("extdata", "kidneyChr21.bed.txt", package="DEGseq")
refFlat    <- system.file("extdata", "refFlatChr21.txt", package="DEGseq")
mapResultBatch <- list(kidneyR1L1)
output <- file.path(tempdir(), "kidneyChr21.bed.exp")
exp <- getGeneExp(mapResultBatch, refFlat=refFlat, output=output)
write.table(exp[30:35,], row.names=FALSE)
cat("output: ", output, "\n")
```

`kidneyChr21.bed` *kidneyChr21.bed*

Description

The reads uniquely mapped to human chromosome 21 obtained from the kidney sample sequenced in Run 1, Lane 1.

References

Marioni,J.C. et al. (2008) RNA-seq: an assessment of technical reproducibility and comparison with gene expression arrays. *Genome Res.*, **18**, 1509-1517.

See Also

`DEGexp`, `DEGseq`, `getGeneExp`, `readGeneExp`, `liverChr21.bed`, `refFlatChr21`.

kidneyChr21Bowtie *kidneyChr21Bowtie*

Description

The reads uniquely mapped to human chromosome 21 obtained from the kidney sample sequenced in Run 1, Lane 1.

References

Marioni,J.C. et al. (2008) RNA-seq: an assessment of technical reproducibility and comparison with gene expression arrays. *Genome Res.*, **18**, 1509-1517.

See Also

DEGexp, DEGseq, getGeneExp, readGeneExp, liverChr21.bed, refFlatChr21.

liverChr21.bed *liverChr21.bed*

Description

The reads uniquely mapped to human chromosome 21 obtained from the liver sample sequenced in Run 1, Lane 2.

References

Marioni,J.C. et al. (2008) RNA-seq: an assessment of technical reproducibility and comparison with gene expression arrays. *Genome Res.*, **18**, 1509-1517.

See Also

DEGexp, DEGseq, getGeneExp, readGeneExp, kidneyChr21.bed, refFlatChr21.

liverChr21Bowtie *liverChr21Bowtie*

Description

The reads uniquely mapped to human chromosome 21 obtained from the liver sample sequenced in Run 1, Lane 2.

References

Marioni,J.C. et al. (2008) RNA-seq: an assessment of technical reproducibility and comparison with gene expression arrays. *Genome Res.*, **18**, 1509-1517.

See Also

DEGexp, DEGseq, getGeneExp, readGeneExp, kidneyChr21.bed, refFlatChr21.

readGeneExp *readGeneExp: read gene expression values to a matrix*

Description

This method is used to read gene expression values from a file to a matrix in R workspace. So that the matrix can be used as input of other packages, such as *edgeR*. The input of the method is a file that contains gene expression values.

Usage

```
readGeneExp(file, geneCol=1, valCol=2, label = NULL, header=TRUE, sep="")
```

Arguments

file	file containing gene expression values.
geneCol	gene id column in file.
valCol	expression value *columns* to be read in the file.
label	label for the columns.
header	a logical value indicating whether the file contains the names of the variables as its first line. See ?read.table.
sep	the field separator character. If sep = "" (the default for read.table) the separator is *white space*, that is one or more spaces, tabs, newlines or carriage returns. See ?read.table.

See Also

getGeneExp, GeneExpExample1000, GeneExpExample5000.

Examples

```
## If the data files are collected in a zip archive, the following
## commands will first extract them to the temporary directory.

geneExpFile <- system.file("extdata", "GeneExpExample1000.txt", package="DEGseq")
exp <- readGeneExp(file=geneExpFile, geneCol=1, valCol=c(7,9,12,15,18,8,10,11,13,16))
exp[30:35,]
```

refFlatChr21 *refFlatChr21*

Description

The gene annotation file includes the annotations of genes on chromosome 21, and is in UCSC refFlat format. See http://genome.ucsc.edu/goldenPath/gbdDescriptionsOld.html#RefFlat.

See Also

DEGseq, DEGexp, kidneyChr21.bed, liverChr21.bed.

samWrapper *samWrapper: A Wrapper (with some modification) of the functions in the package samr to identify differentially expressed genes for the RNA-seq data from two groups of paired or unpaired samples.*

Description

This function is a wrapper of the functions in *samr*. It is used to identify differentially expressed genes for two sets of samples with multiple replicates or two groups of samples from different individuals (e.g. disease samples vs. control samples). For the advanced users, please see *samr* http://cran.r-project.org/web/packages/samr/index.html for detail.

Usage

```
samWrapper(geneExpFile1, geneCol1=1, expCol1=2, measure1=rep(1, length(expCol1))
           geneExpFile2, geneCol2=1, expCol2=2, measure2=rep(2, length(expCol2))
           header=TRUE, sep="", paired=FALSE, s0=NULL, s0.perc=NULL, nperms=100,
           testStatistic=c("standard","wilcoxon"), max.qValue=1e-3, min.foldchan
           logged2=FALSE, output)
```

Arguments

geneExpFile1	file containing gene expression values for group1.
geneCol1	gene id column in geneExpFile1.
expCol1	expression value *columns* in geneExpFile1. See the example.
measure1	numeric vector of outcome measurements for group1. like c(1,1,1...) when paired=FALSE, or like c(-1,-2,-3,...) when paired=TRUE.
geneExpFile2	file containing gene expression values for group2.
geneCol2	gene id column in geneExpFile2.
expCol2	expression value *columns* in geneExpFile2. See the example.
measure2	numeric vector of outcome measurements for group2. like c(2,2,2...) when paired=FALSE, or like c(1,2,3,...) when paired=TRUE.
header	a logical value indicating whether geneExpFile1 and geneExpFile2 contain the names of the variables as its first line. See ?read.table.
sep	the field separator character. If sep = "" (the default for read.table) the separator is *white space*, that is one or more spaces, tabs, newlines or carriage returns. See ?read.table.
paired	a logical value indicating whether the samples are paired.
s0	exchangeability factor for denominator of test statistic; Default is automatic choice.
s0.perc	percentile of standard deviation values to use for s0; default is automatic choice; -1 means s0=0 (different from s0.perc=0, meaning s0=zeroeth percentile of standard deviation values= min of sd values.
nperms	number of permutations used to estimate false discovery rates.

`testStatistic`
: test statistic to use in two class unpaired case. Either "`standard`" (t-statistic) or "`wilcoxon`" (Two-sample wilcoxon or Mann-Whitney test). recommend "`standard`".

`max.qValue`
: the max qValue desired; shoube be <1.

`min.foldchange`
: the minimum fold change desired; should be >1. default is zero, meaning no fold change criterion is applied.

`logged2`
: a logical value indicating whether the expression values are logged2.

`output`
: the output file.

References

Tusher,V., and et al. (2001): Significance analysis of microarrays applied to the ionizing radiation response *PNAS* **98**, 5116-5121.

Tibshirani,R, and et al.: samr http://cran.r-project.org/web/packages/samr/index.html.

A more complete description is given in the SAM manual at http://www-stat.stanford.edu/~tibs/SAM.

See Also

DEGexp, DEGseq, GeneExpExample1000, GeneExpExample5000.

Examples

```
## If the data files are collected in a zip archive, the following
## commands will first extract them to the temporary directory.

geneExpFile <- system.file("extdata", "GeneExpExample1000.txt", package="DEGseq")
set.seed(100)
geneExpFile1 <- geneExpFile
geneExpFile2 <- geneExpFile
output <- file.path(tempdir(), "samWrapperOut.txt")
exp <- readGeneExp(file=geneExpFile, geneCol=1, valCol=c(7,9,12,15,18))
exp[30:35,]
exp <- readGeneExp(file=geneExpFile, geneCol=1, valCol=c(8,10,11,13,16))
exp[30:35,]
samWrapper(geneExpFile1=geneExpFile1, geneCol1=1, expCol1=c(7,9,12,15,18), measure1=c(-
           geneExpFile2=geneExpFile2, geneCol2=1, expCol2=c(8,10,11,13,16), measure2=c(
           nperms=100, min.foldchange=2, max.qValue=1e-4, output=output, paired=TRUE)
cat("output:", output, "\n")
```

Index