How to use the DEGseq Package

Likun Wang[1,2] and Xi Wang[1].

December 31, 2009

[1]MOE Key Laboratory of Bioinformatics and Bioinformatics Division, TNLIST /Department of Automation, Tsinghua University.
[2]College of Computer Science and Technology, Jilin University.

wanglikun@tsinghua.edu.cn

Contents

1 Introduction

This document provides a discussion of the functions in the package *DEGseq*. *DEGseq* is a free R package for identifying differentially expressed genes from RNA-seq data. The input of *DEGseq* is uniquely mapped reads from RNA-seq data with a gene annotation of the corresponding genome, or gene (or transcript isoform) expression values provided by other programs. The output of *DEGseq* includes a text file and an XHTML summary page. The text file contains the gene expression values for the samples, a *P-value* and two kinds of

Q-values which are calculated by the methods described in Benjamini and Hochberg (1995) and Storey and Tibshirani (2003) for each gene to denote its expression difference between libraries.

We also provided a function `samWrapper` using the method as described in SAM (Tusher and et al., 2001) which can be applied to compare two sets of samples with multiple replicates or two groups of samples from different individuals (e.g. disease samples, case vs. control).

The *DEGseq* package employs library *qvalue* and *samr*, which must be installed in advance.

2 Getting started

To load the *DEGseq* package, type `library(DEGseq)`. Total six methods are presented in this package. They are `DEGexp`, `DEGseq`, `samWrapper`, `getGeneExp` and `readGeneExp`.

3 Methods

3.1 MA-plot-based method with random sampling model

Current observations suggest that typically RNA-seq experiments have low background noise and the Poisson model fits data well. In such cases, users could directly pool the technical replicates for each sample together to get higher sequencing depth and detect subtle gene expression changes.

Jiang and et al. (2009) modeled RNA sequencing as a random sampling process, in which each read is sampled independently and uniformly from every possible nucleotides in the sample. Based on this model, the number of reads coming from a gene (or transcript isoform) follows a binomial distribution (and could be approximated by a Poisson distribution). Using the statistical model, we proposed a novel method based on the MA-plot, which is a statistical analysis tool having been widely used to detect and visualize intensity-dependent ratio of microarray data (Yang and et al., 2002).

Let C_1 and C_2 denote the counts of reads mapped to a specific gene obtained from two samples with $C_i \sim binomial(n_i, p_i), i = 1, 2$, where n_i denotes the total number of mapped reads and p_i denotes the probability of a read coming from that gene. We define $M = log_2^{C_1} - log_2^{C_2}$, and $A = (log_2^{C_1} + log_2^{C_2})/2$. We assume that C_1 and C_2 are independent. Let $X = log_2^{C_1}$ and $Y = log_2^{C_2}$, hence $M = X - Y$ and $A = (X + Y)/2$. We can prove that X and Y follow normal distributions approximately (when n_i is large enough), denote

$$X \to N(log_2(n_1 p_1), (\frac{1 - p_1}{n_1 p_1})(log_2^e)^2) = N(\mu_X, \sigma_X^2) \tag{1}$$

$$Y \to N(log_2(n_2 p_2), (\frac{1 - p_2}{n_2 p_2})(log_2^e)^2) = N(\mu_Y, \sigma_Y^2) \tag{2}$$

Based on the assumption that C_1 and C_2 are independent (so X and Y are independent), the distributions of M and A can be obtained:

$$M \sim N(\mu_X - \mu_Y, \sigma_X^2 + \sigma_Y^2) = N(\mu_M, \sigma_M^2) \tag{3}$$

$$A \sim N(\frac{1}{2}(\mu_X + \mu_Y), \frac{1}{4}(\sigma_X^2 + \sigma_Y^2)) = N(\mu_A, \sigma_A^2) \quad (4)$$

Based on formulas (3) and (4), the conditional distribution of M given that $A = a$ can be obtained:

$$M|(A = a) \sim N(\mu_M + \rho\frac{\sigma_M}{\sigma_A}(a - \mu_A), \sigma_M^2(1 - \rho^2)),$$
$$\rho = \frac{Cov(M, A)}{\sigma_M \sigma_A} = \frac{\sigma_X^2 - \sigma_Y^2}{\sigma_X^2 + \sigma_Y^2}.$$

Thus,

$$\begin{aligned} E(M|A = a) &= \mu_M + \rho\frac{\sigma_M}{\sigma_A}(a - \mu_A) \\ &= \mu_X - \mu_Y + 2\frac{\sigma_X^2 - \sigma_Y^2}{\sigma_X^2 + \sigma_Y^2}(a - \frac{1}{2}(\mu_X + \mu_Y)). \end{aligned}$$

and

$$\begin{aligned} Var(M|A = a) &= \sigma_M^2(1 - \rho^2) \\ &= 4\frac{\sigma_X^2\sigma_Y^2}{\sigma_X^2 + \sigma_Y^2}. \end{aligned}$$

For gene g with ($A = a$, $M = m$) on the MA-plot of two samples, we do the hypothesis test $H_0 : p_1 = p_2 = p$ versus $H_1 : p_1 \neq p_2$. Based on above deduction,

$$\mu_A = \frac{1}{2}(\mu_X + \mu_Y) = \frac{1}{2}log_2(n_1 n_2 p^2).$$

Thus,

$$p = \sqrt{2^{2\mu_A}/(n_1 n_2)}.$$

Use a as an estimate of μ_A then

$$\hat{p} = \sqrt{2^{2a}/(n_1 n_2)}.$$

So the estimates of $E(M|A = a)$ and $Var(M|A = a)$ are

$$\widehat{E}(M|A = a) = log_2(n_1) - log_2(n_2),$$

and

$$\widehat{Var}(M|A = a) = \frac{4(1 - \sqrt{2^{2a}/(n_1 n_2)})(log_2^e)^2}{(n_1 + n_2)\sqrt{2^{2a}/(n_1 n_2)}}.$$

Then use the two estimates to calculate a *Z-score* for the gene g with ($A = a$, $M = m$), and convert it to a two-sided *P-value* which is used to indicate whether gene g is differentially expressed or not.

$$Z - score = \frac{|m - \widehat{E}(M|A = a)|}{\sqrt{\widehat{Var}(M|A = a)}}.$$

Given a *Z-score* threshold, take four as an example, the two lines with the following equations are used to indicate the four-fold local standard deviation of M according to the random sampling model:

$$\begin{aligned} m_1 &= \widehat{E}(M|A=a) + 4 \cdot \sqrt{\widehat{Var}(M|A=a)} \\ &= log_2(n_1) - log_2(n_2) + 4 \cdot \sqrt{\frac{4(1-\sqrt{2^{2a}/(n_1 n_2)})(log_2^e)^2}{(n_1+n_2)\sqrt{2^{2a}/(n_1 n_2)}}} \end{aligned}$$

$$\begin{aligned} m_2 &= \widehat{E}(M|A=a) - 4 \cdot \sqrt{\widehat{Var}(M|A=a)} \\ &= log_2(n_1) - log_2(n_2) - 4 \cdot \sqrt{\frac{4(1-\sqrt{2^{2a}/(n_1 n_2)})(log_2^e)^2}{(n_1+n_2)\sqrt{2^{2a}/(n_1 n_2)}}} \end{aligned}$$

We call the lines obtained by above equations *theoretical* four-fold local standard deviations lines. Please see Wang and et al. (2009) for detail.

3.2 MA-plot-based method with technical replicates

To estimate the noise level of genes with different intensity, and identify gene expression difference in different sequencing libraries, we proposed another method which is also based on the MA-plot. Here M is the Y-axis and represents the intensity ratio, and A is the X-axis and represents the average intensity for each transcript. To estimate the random variation, we first draw a MA-plot using two technical replicates (e.g. two sequencing lanes) from the same library. A sliding window (each window includes 1% points of the MA-plot) is applied to scan the MA-plot along the A-axis (average intensity). For the window which is centered at $A = a$, the local variation of M conditioned on $A = a$ is estimated by all the M values of the transcripts in the window. And a smoothed estimate of the intensity-dependent noise level (local variation of M) is estimated by lowess regression among the windows, and converted to the standard deviation, under the assumption of normal distribution. The local standard deviations σ_a of M conditioned on $A = a$ were then used to compare the observed difference between two different libraries. Next, we draw a second MA-plot for the data from two different libraries. For each transcript g with ($A = a_g$, $M = m_g$) on the MA-plot, a *Z-score* $= |m_g - \mu_g|/\sigma_g$ is calculated to evaluate whether this transcript is differentially expressed, where μ_g is the local mean of M and σ_g is the local standard deviation of M conditioned on $A = a_g$ estimated by technical replicates. Finally, a *P-value* is assigned to this gene according to the *Z-score* under the assumption of normal distribution. Please see Wang and et al. (2009) for detail.

4 Data

The test RNA-seq data are from Marioni and et al. (2008). In their research, the RNA samples from human liver and kidney were analyzed using the Illumina Genome Analyzer sequencing platform. Each sample was sequenced in seven lanes, split across two runs of the machine. The raw data are available in the NCBI short read archive with accession number `SRA000299`. Please see Marioni and et al. (2008) for more details.

5 Examples

5.1 Example for DEGexp

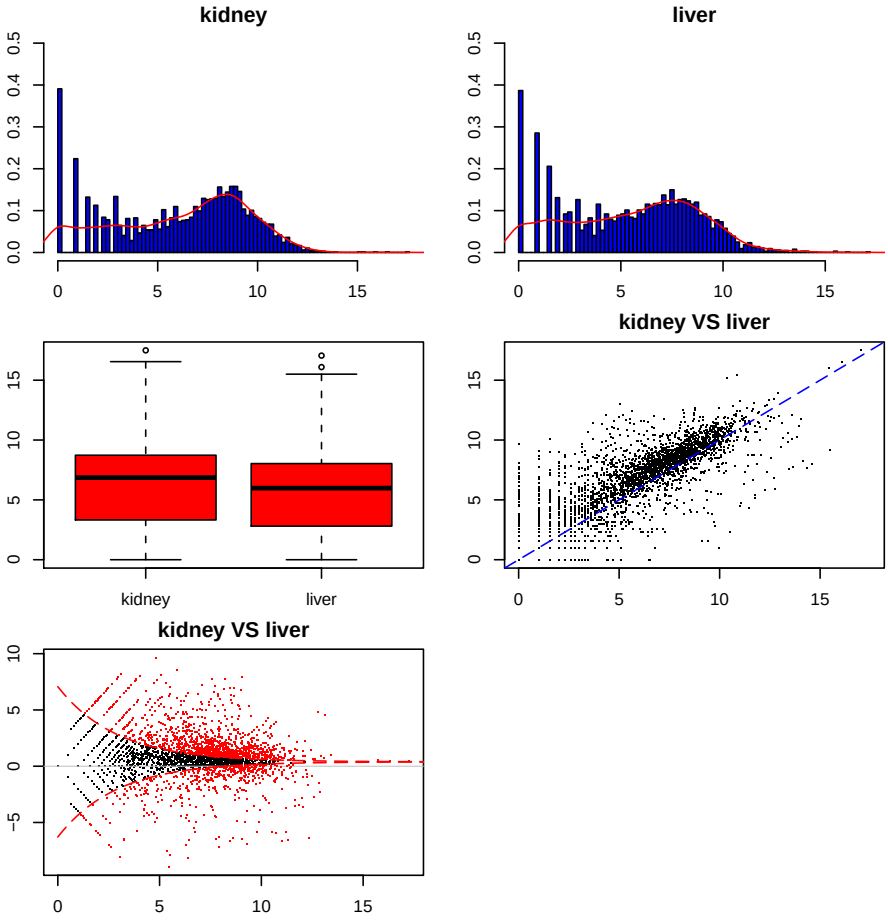
If users already have the gene expression values (such as the number of reads mapped to each gene), this method can be used to identify differentially expressed genes between two samples with or without multiple technical replicates directly. In the package, there are test data for this method. The file GeneExpExample5000.txt includes the first 5000 lines in SupplementaryTable2.txt which is a supplementary file for Marioni and et al. (2008). In this file, each line includes the count of reads mapped to a gene for 14 lanes respectively (7 lanes for kidney and 7 lanes for liver). In the following examples, we only use the data sequenced at a concentration of 3 pM (five lanes for each sample). If the data files are collected in a zip archive, the following commands will first extract them from the archive to the temporary directory.

```
> library(DEGseq)
> geneExpFile <- system.file("data", "GeneExpExample5000.txt",
+     package = "DEGseq")
> if (geneExpFile == "") {
+     zipFile <- system.file("data", "Rdata.zip", package = "DEGseq")
+     if (zipFile != "") {
+         unzip(zipFile, "GeneExpExample5000.txt", exdir = tempdir())
+         geneExpFile <- file.path(tempdir(), "GeneExpExample5000.txt")
+     }
+ }
```

To identify differentially expressed genes between the two samples (kidney and liver), we first used the method `MARS`: MA-plot-based method with Random Sampling model. Five report graphs for the two samples will be shown following the example commands.

```
> layout(matrix(c(1, 2, 3, 4, 5, 6), 3, 2, byrow = TRUE))
> par(mar = c(2, 2, 2, 2))
> DEGexp(geneExpFile1 = geneExpFile, expCol1 = c(7, 9, 12, 15,
+     18), groupLabel1 = "kidney", geneExpFile2 = geneExpFile,
+     expCol2 = c(8, 10, 11, 13, 16), groupLabel2 = "liver", method = "MARS")

Please wait...

geneExpFile1:  /tmp/Rinst2035059849/DEGseq/data/GeneExpExample5000.txt
gene id column in geneExpFile1:  1
expression value column(s) in geneExpFile1: 7 9 12 15 18
total number of reads uniquely mapped to genome obtained from sample1: 345504 354981 334557

geneExpFile2:  /tmp/Rinst2035059849/DEGseq/data/GeneExpExample5000.txt
gene id column in geneExpFile2:  1
expression value column(s) in geneExpFile2: 8 10 11 13 16
total number of reads uniquely mapped to genome obtained from sample2: 274430 274486 264999
```

```
method to identify differentially expressed genes:  MARS
pValue threshold: 0.001
output directory: none

The outputDir is not specified!
Please wait ...
Identifying differentially expressed genes ...
Please wait patiently ...
output ...
The results can be observed in directory:  none
```

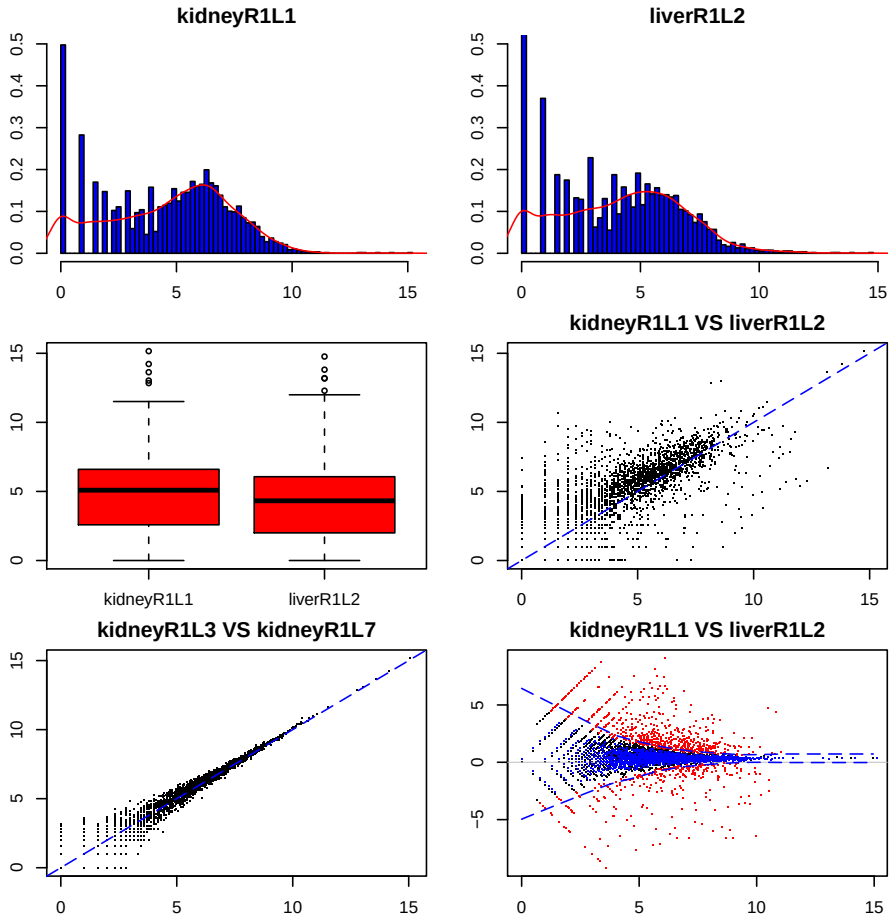


The red points in the last graph (MA-plot) are the genes identified as differentially expressed. If the `outputDir` is specified, a text file and an XHTML summary page will be generated. These files can be found in the output directory.

Next, we performed the function `DEGexp` with the method `MATR`: MA-plot-based method with technical replicates.

```
> layout(matrix(c(1, 2, 3, 4, 5, 6), 3, 2, byrow = TRUE))
> par(mar = c(2, 2, 2, 2))
```

```
> DEGexp(geneExpFile1 = geneExpFile, expCol1 = 7, groupLabel1 = "kidneyR1L1",
+     geneExpFile2 = geneExpFile, expCol2 = 8, groupLabel2 = "liverR1L2",
+     replicate1 = geneExpFile, expColR1 = 9, replicate2 = geneExpFile,
+     expColR2 = 12, replicateLabel1 = "kidneyR1L3", replicateLabel2 = "kidneyR1L7",
+     method = "MATR")
```



The red points in the last graph (MA-plot) are the genes identified as differentially expressed. The blue points are from the replicates (kidneyR1L3 and kidneyR1L7), and the blue lines show the four-fold local standard deviation of M estimated by the two technical replicates.

5.2 Example for DEGseq

The method `DEGseq` takes uniquely mapped reads from RNA-seq data with a gene annotation as input. This function first counts the number of reads mapped to each gene for samples with or without multiple technical replicates. And then it invokes `DEGexp` to identify significant genes.

```
> kidneyR1L1 <- system.file("data", "kidneyChr21.bed.txt", package = "DEGseq")
> liverR1L2 <- system.file("data", "liverChr21.bed.txt", package = "DEGseq")
```

```
> refFlat <- system.file("data", "refFlatChr21.txt", package = "DEGseq")
> mapResultBatch1 <- c(kidneyR1L1)
> mapResultBatch2 <- c(liverR1L2)
> outputDir <- file.path(tempdir(), "DEGseqExample")
> DEGseq(mapResultBatch1, mapResultBatch2, fileFormat = "bed",
+     refFlat = refFlat, outputDir = outputDir, method = "LRT")

Please wait...

mapResultBatch1:
         /tmp/Rinst2035059849/DEGseq/data/kidneyChr21.bed.txt
mapResultBatch2:
         /tmp/Rinst2035059849/DEGseq/data/liverChr21.bed.txt
file format: bed
refFlat:  /tmp/Rinst2035059849/DEGseq/data/refFlatChr21.txt
Ignore the strand information when count the reads mapped to genes!
Count the number of reads mapped to each gene ...
This will take several minutes, please wait patiently!
Please wait...

SampleFiles:
        /tmp/Rinst2035059849/DEGseq/data/kidneyChr21.bed.txt
Count the number of reads mapped to each gene.
This will take several minutes.
Please wait ...
total 259 unique genes

processed 0 reads (kidneyChr21.bed.txt)
processed 10000 reads (kidneyChr21.bed.txt)
processed 20000 reads (kidneyChr21.bed.txt)
processed 30000 reads (kidneyChr21.bed.txt)
processed 34304 reads (kidneyChr21.bed.txt)
total used 0.330000 seconds!
Please wait...

SampleFiles:
        /tmp/Rinst2035059849/DEGseq/data/liverChr21.bed.txt
Count the number of reads mapped to each gene.
This will take several minutes.
Please wait ...
total 259 unique genes

processed 0 reads (liverChr21.bed.txt)
processed 10000 reads (liverChr21.bed.txt)
processed 20000 reads (liverChr21.bed.txt)
```

```
processed 30000 reads (liverChr21.bed.txt)
processed 30729 reads (liverChr21.bed.txt)
total used 0.290000 seconds!
Please wait...

geneExpFile1:  /tmp/RtmpjLr6sp/DEGseqExample/group1.exp
gene id column in geneExpFile1:  1
expression value column(s) in geneExpFile1: 2
total number of reads uniquely mapped to genome obtained from sample1: 34304

geneExpFile2:  /tmp/RtmpjLr6sp/DEGseqExample/group2.exp
gene id column in geneExpFile2:  1
expression value column(s) in geneExpFile2: 2
total number of reads uniquely mapped to genome obtained from sample2: 30729

method to identify differentially expressed genes:  LRT
pValue threshold: 0.001
output directory: /tmp/RtmpjLr6sp/DEGseqExample

Please wait ...
Identifying differentially expressed genes ...
Please wait patiently ...
output ...

Done ...
The results can be observed in directory:  /tmp/RtmpjLr6sp/DEGseqExample
```

5.3 Example for samWrapper

To compare two sets of samples with multiple replicates or two groups of samples from different individuals (e.g. disease samples vs. control samples), we provided a method which employs the methods in package *samr*. The strategy used in *samr* was first described in Tusher and et al. (2001), and is used for significance analysis of microarrays.

```
> geneExpFile <- system.file("data", "GeneExpExample1000.txt",
+     package = "DEGseq")
> set.seed(100)
> geneExpFile1 <- geneExpFile
> geneExpFile2 <- geneExpFile
> output <- file.path(tempdir(), "samWrapperOut.txt")
> expCol1 = c(7, 9, 12, 15, 18)
> expCol2 = c(8, 10, 11, 13, 16)
> measure1 = c(-1, -2, -3, -4, -5)
> measure2 = c(1, 2, 3, 4, 5)
> samWrapper(geneExpFile1 = geneExpFile1, geneCol1 = 1, expCol1 = expCol1,
+     measure1 = measure1, geneExpFile2 = geneExpFile2, geneCol2 = 1,
```

```
+     expCol2 = expCol2, measure2 = measure2, nperms = 100, min.foldchange = 2,
+     max.qValue = 1e-04, output = output, paired = TRUE)
```

For the advanced users, please see samr `http://cran.r-project.org/web/packages/samr/index.html` for detail.

5.4 Example for getGeneExp

This method is used to count the number of reads mapped to each gene for one sample. The sample can have multiple technical replicates. The input of this method is the uniquely mapped reads with a gene annotation. And the output is a text file containing gene expression values for the sample. For example,

```
> kidneyR1L1 <- system.file("data", "kidneyChr21.bed.txt", package = "DEGseq")
> refFlat <- system.file("data", "refFlatChr21.txt", package = "DEGseq")
> mapResultBatch <- c(kidneyR1L1)
> output <- file.path(tempdir(), "kidneyChr21.bed.exp")
> getGeneExp(mapResultBatch, refFlat = refFlat, output = output)

Please wait...

SampleFiles:
        /tmp/Rinst2035059849/DEGseq/data/kidneyChr21.bed.txt
Count the number of reads mapped to each gene.
This will take several minutes.
Please wait ...
total 259 unique genes

processed 0 reads (kidneyChr21.bed.txt)
processed 10000 reads (kidneyChr21.bed.txt)
processed 20000 reads (kidneyChr21.bed.txt)
processed 30000 reads (kidneyChr21.bed.txt)
processed 34304 reads (kidneyChr21.bed.txt)
total used 0.340000 seconds!

> exp <- readGeneExp(file = output, geneCol = 1, valCol = c(2,
+     3), label = c("raw count", "RPKM"))
> exp[30:32, ]

          raw count    RPKM
C21orf131         0   0.000
C21orf15          0   0.000
C21orf2          51 665.789
```

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

5.5 Example for readGeneExp

This function is used to read gene expression values from a file to a matrix in R workspace. For example,

```
> geneExpFile <- system.file("data", "GeneExpExample1000.txt",
+     package = "DEGseq")
> exp <- readGeneExp(file = geneExpFile, geneCol = 1, valCol = c(7,
+     9, 12, 15, 18, 8, 10, 11, 13, 16))
> exp[30:32, ]
```

```
                R1L1Kidney R1L3Kidney R1L7Kidney R2L2Kidney R2L6Kidney
ENSG00000188976         73         77         68         70         82
ENSG00000187961         15         15         13         12         15
ENSG00000187583          1          1          3          0          3
                R1L2Liver R1L4Liver R1L6Liver R1L8Liver R2L3Liver
ENSG00000188976        34        56        45        55        42
ENSG00000187961         8        13        11        12        20
ENSG00000187583         0         1         0         0         2
```

6 The relation between DEGseq and edgeR

The package *edgeR* implements the method based on negative binominal distribution to model overdispersion relative to Poisson for digital gene expression data with small replicates (Robinson and et al., 2007). The methods in *edgeR* first estimate the dispersion and then construct an exact test similar to the Fisher's exact test for contingency tables but replacing the hypergeometric probabilities with negative binomial. According to current published RNA-seq data, typically this technology has low instrument noise and the Poisson model fits data well. In this situation, Robinson and et al. (2009) recommended users perform the methods with the dispersion parameter ϕ set to 0 (as $\phi \to 0$, the negative binominal distribution reduces to the Poisson).

Some methods in *DEGseq* are also based on the Poisson model. But the MA-plot-based methods employ MA-plot to identify and visualize significant genes. Besides, our package can help users calculate the count of reads mapped to each gene with the gene annotation file in UCSC refFlat format. The methods in the two packages complement each other. Both the packages can be used to identify significant genes from RNA-seq data.

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 et al. (2009). Statistical inferences for isoform expression in RNA-Seq. *Bioinformatics*, 25:1026–1032.

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

Robinson, M. D. and et al. (2007). Moderated statistical tests for assessing differences in tag abundance. *Bioinformatics*, 23:2881–2887.

Robinson, M. D. and et al. (2009). edgeR: Empirical analysis of digital gene expression data in R. *Bioconductor*.

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

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

Wang, L. K. and et al. (2009). DEGseq: an R package to identify differentially expressed genes from RNA-seq data. *Bioinformatics*, doi:10.1093/bioinformatics/btp612.

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