metaSeq: Meta-analysis of RNA-seq count data

Koki Tsuyuzaki[1], and Itoshi Nikaido[2].

May 1, 2024

[1]Department of Medical and Life Science, Tokyo University of Science.
[2]Bioinformatics Research Unit, Advanced Center for Computing and Communication, RIKEN.

k.t.the-answer@hotmail.co.jp

Contents

1 Introduction

This document provides the way to perform meta-analysis of RNA-seq data using *metaSeq* package. Meta-analysis is a attempt to integrate multiple data in different studies and retrieve much reliable and reproducible result. In transcriptome study, the goal of analysis may be differentially expressed genes (DEGs). In our package, the probability of one-sided *NOISeq* [1] is applied in each study. This is because the numbers of reads are often different depending on its study and *NOISeq* is robust method against its difference (see the next section). By meta-analysis, genes which differentially expressed in many studies are detected as DEGs.

2 RSE: Read-Size Effect

In many cases, the number of reads are depend on study. For example, here we prepared multiple RNA-Seq count data designed as Breast Cancer cell lines vs Normal cells measured in 4 different studies (this data is also accessible by **data**(**BreastCancer**)).

ID in this vignette	Accession (SRA / ERA Accession)	Experimental Design
StudyA	SRP008746	Breast Cancer (n=3) vs Normal (n=2)
StudyB	SRP006726	Breast Cancer (n=1) vs Normal (n=1)
StudyC	SRP005601	Breast Cancer (n=7) vs Normal (n=1)
StudyD	ERP000992	Breast Cancer (n=2) vs Normal (n=1)

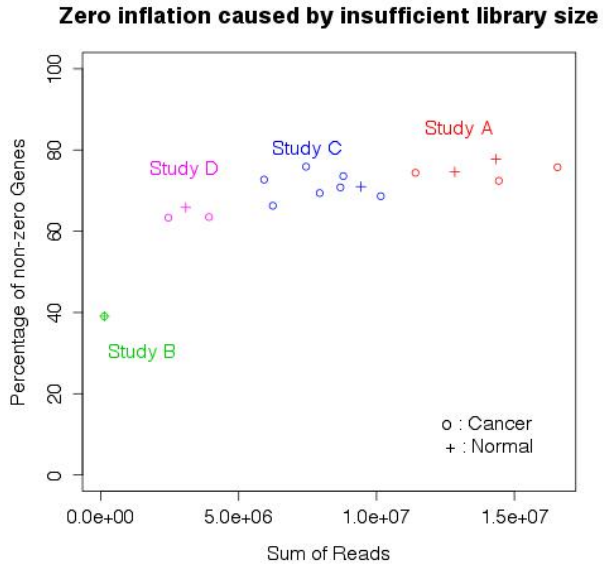

Figure 1: Difference of the number of reads

As shown in the figure 1, the number of reads in StudyA, B, C, and D are relatively different. Generally, statistical test is influenced by the number of reads; the more the number of reads is large, the more the statistical tests are tend to be significant (see the next section). Therefore, in meta-analysis of RNA-seq data, data may be suffered from this bias. Here we call this bias as RSE (Read Size Effect).

3 Robustness against RSE

In the point of view of robustness against RSE, we evaluated five widely used method in RNA-seq; *DESeq* [2], *edgeR* [3], *baySeq* [4], and *NOISeq* [1]. Here we used only StudyA data. All counts in the matrix are repeatedly down-sampled in accordance with distributions of binomial (the probability equals 0.5). 1 (original), 1/2, 1/4, 1/8, 1/16, and 1/32-fold data are prepared as low read size situation. In each read size, four methods are conducted (figure 2.A, this data is also accessible by **data**(**StudyA**) and **data**(**pvals**)), then we focussed on how top500 genes of original data in order of significance will change its members, influenced by low read size (figure 2.B).

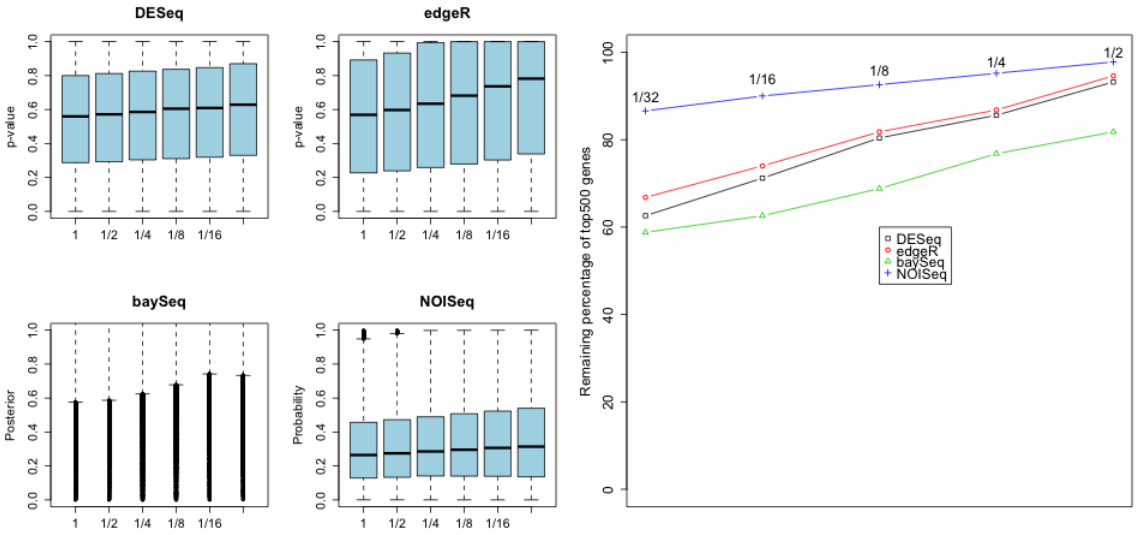

Figure 2: A(left): RSE in each RNA-Seq method, B(right): Top 500 genes in order of significance

Ideal method will returns same result regardless of read size, because same data was used. As shown in figure 2, *NOISeq* is not almost affected by the number of reads and robustlly detects same genes as DEGs. Therefore, we concluded that *NOISeq* is suitable method at least in the point of view of meta-analysis. Note that probability of *NOISeq* is not equal to p-value; it is the probability that a gene is differentially expressed [1]. Our package integrates its probability by Fisher's method [5] or Stouffer's method (inverse normal method) [6]. In regard to Stouffer's method, weighting by the number of replicates (sample size) is used.

4 Getting started

At first, install and load the *metaSeq* and *snow*.

```
> library("metaSeq")
> library("snow")
```

The RNA-seq expression data in breast cancer cell lines and normal cells is prepared. The data is measured from 4 different studies.The data is stored as a matrix (23368 rows × 18 columns).

```
> data(BreastCancer)
```

We need to prepare two vectors. First vector is for indicating the experimental condition (e.g., 1: Cancer, 2: Normal) and second one is for indicating the source of data (e.g., A: StudyA, B: StudyB, C: StudyC, D: StudyD).

```
> flag1 <- c(1,1,1,0,0, 1,0, 1,1,1,1,1,1,1,0, 1,1,0)
> flag2 <- c("A","A","A","A","A", "B","B", "C","C","C","C","C","C","C","C", "D","D","D")
```

Then, we use **meta.readData** to create R object for **meta.oneside.noiseq**.

```
> cds <- meta.readData(data = BreastCancer, factor = flag1, studies = flag2)
```

Onesided-NOISeq is performed in each studies and each probabilities are summalized as a member of list object.

```
> ## This is very time consuming step.
> # cl <- makeCluster(4, "SOCK")
> # result <- meta.oneside.noiseq(cds, k = 0.5, norm = "tmm", replicates = "biological",
> #  factor = flag1, conditions = c(1, 0), studies = flag2, cl = cl)
> # stopCluster(cl)
>
> ## Please load pre-calculated result (Result.Meta)
> ## by data function instead of scripts above.
> data(Result.Meta)
> result <- Result.Meta
```

Fisher's method and Stouffer's method can be applied to the result of **meta.oneside.noiseq**.

```
> F <- Fisher.test(result)
> S <- Stouffer.test(result)
```

These outputs are summalized as list whose length is 3. First member is the probability which means a gene is upper-regulated genes, and Second member is lower-regulated genes. Weight in each study is also saved as its third member (weight is used only by Stouffer's method).

```
> head(F$Upper)

1/2-SBSRNA4        A1BG    A1BG-AS1        A1CF       A2LD1
  0.3842542   0.5316118   0.5325544          NA   0.1358559
       A2M
  0.2252807

> head(F$Lower)

1/2-SBSRNA4        A1BG    A1BG-AS1        A1CF       A2LD1
  0.8420357   0.6078896   0.4047202          NA   0.3661371
       A2M
  0.6197968

> F$Weight

Study 1 Study 2 Study 3 Study 4
      5       2       8       3

> head(S$Upper)

1/2-SBSRNA4        A1BG    A1BG-AS1        A1CF       A2LD1
  0.3709297   0.2663748   0.2711745          NA   0.2957139
       A2M
  0.2996707

> head(S$Lower)

1/2-SBSRNA4        A1BG    A1BG-AS1        A1CF       A2LD1
  0.6290703   0.7336252   0.7288255          NA   0.7042861
       A2M
  0.7003293

> S$Weight

Study 1 Study 2 Study 3 Study 4
      5       2       8       3
```

Generally, by meta-analysis, detection power will improved and much genes are detected as DEGs.

Method	Study	Number of DEGs
NOISeq	A	86
NOISeq	B	563
NOISeq	C	99
NOISeq	D	210
NOISeq	A, B, C, D (not meta-analysis)	21
metaSeq (Fisher, Upper)	A, B, C, D	407
metaSeq (Fisher, Lower)	A, B, C, D	1483
metaSeq (Stouffer, Upper)	A, B, C, D	116
metaSeq (Stouffer, Lower)	A, B, C, D	2271

5 Meta-analysis by non-NOISeq method

For some reason, we may want to use non-NOISeq method like *DESeq*, *edgeR*, or even cuffdiff [7]. We prepared other.oneside.noiseq as optional function for such methods. Returned object can be directlly applied to **Fisher.test** and **Stouffer.test**.

We have to prepare at least 2 matrix filled with p-value or probability. First matrix is for upper-regulated genes between control group and treatment group. On the other hand, second matrix is for lower-regulated genes. As optional parameter, weight in each study is also avilable. Weight is need for Stouffer's method but not necessary for Fisher's method.

```
> ## Assume this matrix as one-sided p-values
> ## generated by non-NOISeq method (e.g., cuffdiff)
> upper <- matrix(runif(300), ncol=3, nrow=100)
> lower <- 1 - upper
> rownames(upper) <- paste0("Gene", 1:100)
> rownames(lower) <- paste0("Gene", 1:100)
> weight <- c(3,6,8)
```

Next, **other.oneside.pvalues** will return a list object for **Fisher.test** or **Stouffer.test** by upper, lower, and weight.

```
> ## other.oneside.pvalues function return a matrix
> ## which can input Fisher.test or Stouffer.test
> result <- other.oneside.pvalues(upper, lower, weight)
```

result above can be applied to **Fisher.test** and **Stouffer.test**.

```
> F <- Fisher.test(result)
> str(F)
```

```
List of 3
 $ Upper : Named num [1:100] 0.11 0.138 0.756 0.999 0.269 ...
  ..- attr(*, "names")= chr [1:100] "Gene1" "Gene2" "Gene3" "Gene4" ...
 $ Lower : Named num [1:100] 0.9611 0.50982 0.22777 0.00121 0.41226 ...
  ..- attr(*, "names")= chr [1:100] "Gene1" "Gene2" "Gene3" "Gene4" ...
 $ Weight: Named num [1:3] 3 6 8
  ..- attr(*, "names")= chr [1:3] "Exp 1" "Exp 2" "Exp 3"
```

```
> head(F$Upper)
```

```
    Gene1     Gene2     Gene3     Gene4     Gene5     Gene6
0.1098219 0.1379658 0.7557401 0.9994833 0.2694677 0.1924877
```

```
> head(F$Lower)
```

```
      Gene1       Gene2       Gene3       Gene4       Gene5
0.961098772 0.509815587 0.227770771 0.001210061 0.412255152
      Gene6
0.887614105
```

```
> F$Weight

Exp 1 Exp 2 Exp 3
    3     6     8

> S <- Stouffer.test(result)
> str(S)

List of 3
 $ Upper : Named num [1:100] 0.0944 0.1801 0.8866 0.9986 0.2213 ...
  ..- attr(*, "names")= chr [1:100] "Gene1" "Gene2" "Gene3" "Gene4" ...
 $ Lower : Named num [1:100] 0.90559 0.81994 0.11345 0.00139 0.7787 ...
  ..- attr(*, "names")= chr [1:100] "Gene1" "Gene2" "Gene3" "Gene4" ...
 $ Weight: Named num [1:3] 3 6 8
  ..- attr(*, "names")= chr [1:3] "Exp 1" "Exp 2" "Exp 3"

> head(S$Upper)

     Gene1      Gene2      Gene3      Gene4      Gene5      Gene6
0.09441094 0.18006068 0.88655007 0.99860989 0.22129813 0.20308645

> head(S$Lower)

      Gene1       Gene2       Gene3       Gene4       Gene5
0.905589059 0.819939315 0.113449935 0.001390108 0.778701873
      Gene6
0.796913546

> S$Weight

Exp 1 Exp 2 Exp 3
    3     6     8
```

6 Setup

This vignette was built on:

```
> sessionInfo()

R version 4.4.0 RC (2024-04-16 r86468 ucrt)
Platform: x86_64-w64-mingw32/x64
Running under: Windows Server 2022 x64 (build 20348)

Matrix products: default


locale:
[1] LC_COLLATE=C
[2] LC_CTYPE=English_United States.utf8
[3] LC_MONETARY=English_United States.utf8
[4] LC_NUMERIC=C
[5] LC_TIME=English_United States.utf8

time zone: America/New_York
tzcode source: internal

attached base packages:
[1] splines   stats     graphics  grDevices utils     datasets
[7] methods   base

other attached packages:
[1] metaSeq_1.45.0      Rcpp_1.0.12         snow_0.4-4
[4] NOISeq_2.49.0       Matrix_1.7-0        Biobase_2.65.0
[7] BiocGenerics_0.51.0

loaded via a namespace (and not attached):
[1] compiler_4.4.0 parallel_4.4.0 tools_4.4.0    grid_4.4.0
[5] lattice_0.22-6
```

References

[1] Tarazona, S. and Garcia-Alcalde, F. and Dopazo, J. and Ferrer, A. and Conesa, A. Genome Research *Differential expression in RNA-seq: A matter of depth*, 21(12): 2213-2223, 2011.

[2] Simon Anders and Wolfgang Huber Genome Biology *Differential expression analysis for sequence count data.*, 11: R106, 2010.

[3] Robinson, M. D. and McCarthy, D. J. and Smyth, G. K. Bioinformatics *edgeR: a Bioconductor package for differential expression analysis of digital gene expression data.*, 26: 139-140, 2010

[4] Thomas J. Hardcastle R package version 1.14.1. *baySeq: Empirical Bayesian analysis of patterns of differential expression in count data.*, 2012.

[5] Fisher, R. A. Statistical Methods for Research Workers, 4th edition, Oliver and Boyd, London, 1932.

[6] Stouffer, S. A. and Suchman, E. A. and DeVinney, L. C. and Star, S. A. and Williams, R. M. Jr. The American Soldier, Vol. 1 - Adjustment during Army Life. Princeton, Princeton University Press, 1949

[7] Trapnell, C. and Williams, B. A. and Pertea, G. and Mortazavi, A. and Kwan, G. and Baren, M. J. and Salzberg, S. L. and Wold, B. J. and Pachter, L. Nature biotechnology *Transcript assembly and quantification by RNA-Seq reveals unannotated transcripts and isoform switching during cell differentiaiton*, 28: 511-515, 2010.