metaSeq: Meta-analysis of RNA-seq count data

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

September 26, 2013

[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)

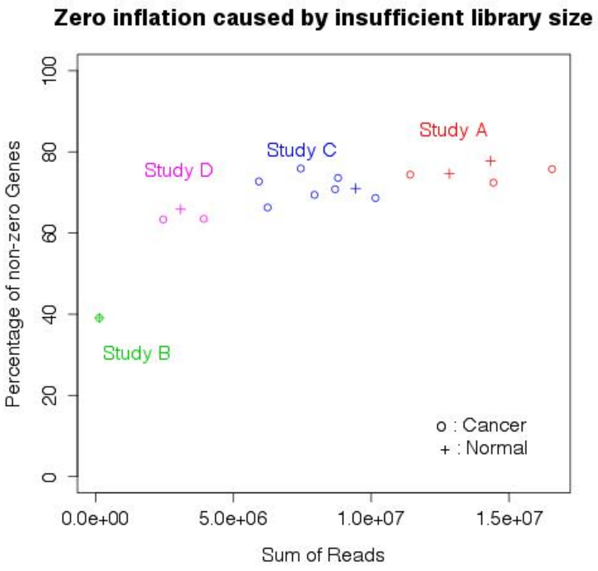

Figure 1: Difference of the number of reads

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).

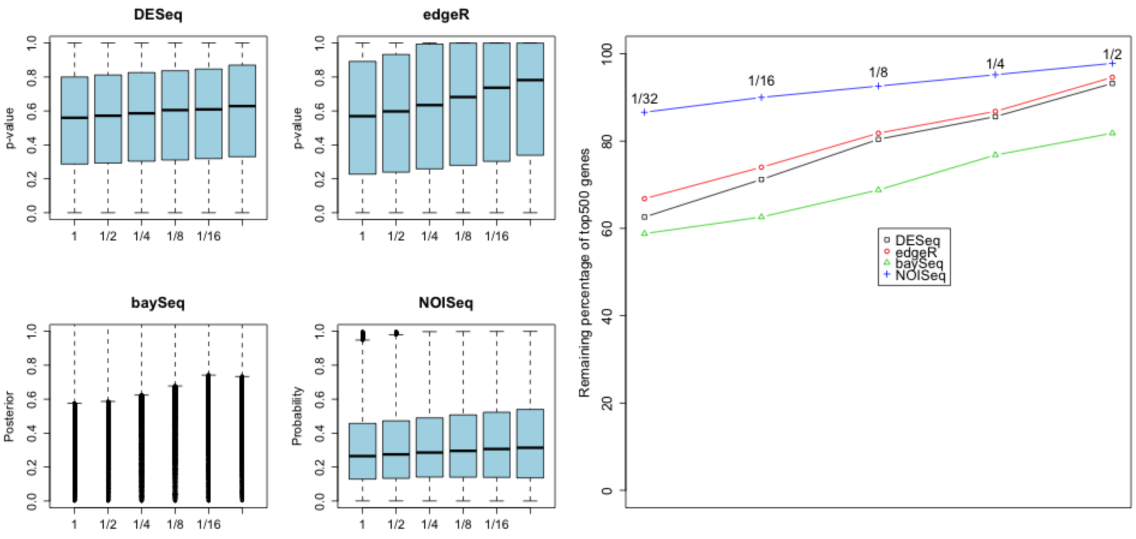

Figure 2: A(left): RSE in each RNA-Seq method, B(right): Top 500 genes in order of significance

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)
```

oneside.noiseq is performed in each studies and each probabilities are summalized as 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 for Fisher.test and Stouffer.test.

```
> ## 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)
> ## other.oneside.pvalues function return a matrix
> ## which can input Fisher.test or Stouffer.test
> result <- other.oneside.pvalues(upper, lower, weight)
> ## Fisher's method (without weighting)
> F <- Fisher.test(result)
> str(F)
```

```
List of 3
 $ Upper : Named num [1:100] 0.16 0.198 0.923 0.984 0.536 ...
  ..- attr(*, "names")= chr [1:100] "Gene1" "Gene2" "Gene3" "Gene4" ...
 $ Lower : Named num [1:100] 0.8022 0.9032 0.0353 0.0424 0.4961 ...
  ..- 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"
```

```
> F
```

```
$Upper
      Gene1       Gene2       Gene3       Gene4       Gene5 
0.159875065 0.197725806 0.923383141 0.983610333 0.535757906 
      Gene6       Gene7       Gene8       Gene9      Gene10 
0.920889635 0.873602558 0.249752446 0.872914948 0.133986607 
     Gene11      Gene12      Gene13      Gene14      Gene15 
0.489512907 0.929058406 0.386765371 0.526438260 0.539368054 
     Gene16      Gene17      Gene18      Gene19      Gene20 
0.951406406 0.003573111 0.107093913 0.898824475 0.479639400 
     Gene21      Gene22      Gene23      Gene24      Gene25 
0.351684130 0.858412973 0.012469969 0.484200452 0.615012944 
     Gene26      Gene27      Gene28      Gene29      Gene30 
0.650816087 0.002317375 0.720109238 0.120809551 0.217776232 
     Gene31      Gene32      Gene33      Gene34      Gene35 
0.298999910 0.770990058 0.242536812 0.776005931 0.259569360 
     Gene36      Gene37      Gene38      Gene39      Gene40 
```

```
0.654146510 0.669378757 0.658020379 0.950435975 0.986829528
     Gene41      Gene42      Gene43      Gene44      Gene45
0.569877232 0.818697098 0.782434052 0.816168221 0.039881958
     Gene46      Gene47      Gene48      Gene49      Gene50
0.674414684 0.754920587 0.574709389 0.335059727 0.234532090
     Gene51      Gene52      Gene53      Gene54      Gene55
0.364972755 0.295187921 0.310368452 0.574041792 0.268403135
     Gene56      Gene57      Gene58      Gene59      Gene60
0.894400010 0.964245797 0.994033989 0.254863274 0.610752114
     Gene61      Gene62      Gene63      Gene64      Gene65
0.584150480 0.825712368 0.470067187 0.405951689 0.775035122
     Gene66      Gene67      Gene68      Gene69      Gene70
0.082620317 0.472499023 0.923546434 0.253279442 0.062322419
     Gene71      Gene72      Gene73      Gene74      Gene75
0.260430114 0.364218219 0.110745718 0.110502416 0.112895333
     Gene76      Gene77      Gene78      Gene79      Gene80
0.884988179 0.271582281 0.901276258 0.520077452 0.686294095
     Gene81      Gene82      Gene83      Gene84      Gene85
0.036507281 0.092900240 0.822198109 0.074884799 0.615155676
     Gene86      Gene87      Gene88      Gene89      Gene90
0.940800216 0.095532212 0.026009877 0.543448896 0.593575464
     Gene91      Gene92      Gene93      Gene94      Gene95
0.294001318 0.771838182 0.715197214 0.338094362 0.743634636
     Gene96      Gene97      Gene98      Gene99     Gene100
0.262948653 0.857703101 0.493551579 0.539288171 0.854332656

$Lower
     Gene1      Gene2      Gene3      Gene4      Gene5      Gene6
0.80218603 0.90318025 0.03533761 0.04236745 0.49611703 0.22547638
     Gene7      Gene8      Gene9     Gene10     Gene11     Gene12
0.05054774 0.63600058 0.35136810 0.92265507 0.61915311 0.18333008
    Gene13     Gene14     Gene15     Gene16     Gene17     Gene18
0.70373015 0.71270159 0.74689506 0.15731706 0.99040684 0.65504398
    Gene19     Gene20     Gene21     Gene22     Gene23     Gene24
0.02818319 0.21365367 0.50589011 0.05636858 0.99443128 0.77975510
    Gene25     Gene26     Gene27     Gene28     Gene29     Gene30
0.63295710 0.57363850 0.99650598 0.36664259 0.86532634 0.50909075
    Gene31     Gene32     Gene33     Gene34     Gene35     Gene36
0.77952507 0.50497485 0.60812779 0.44912196 0.01646054 0.56023039
    Gene37     Gene38     Gene39     Gene40     Gene41     Gene42
0.02227271 0.46501104 0.10560902 0.05586979 0.57304558 0.15971253
    Gene43     Gene44     Gene45     Gene46     Gene47     Gene48
0.04204417 0.28877034 0.71914405 0.38650333 0.06106894 0.46081302
    Gene49     Gene50     Gene51     Gene52     Gene53     Gene54
0.51485659 0.74974396 0.55069067 0.77656983 0.45801632 0.59862066
```

```
    Gene55     Gene56     Gene57     Gene58     Gene59     Gene60 
0.80176012 0.12748068 0.11715978 0.02026346 0.91313732 0.11299892 
    Gene61     Gene62     Gene63     Gene64     Gene65     Gene66 
0.37745724 0.40012238 0.54323458 0.36897504 0.33195765 0.60527135 
    Gene67     Gene68     Gene69     Gene70     Gene71     Gene72 
0.32301031 0.02604410 0.79227970 0.46648852 0.80552857 0.34054517 
    Gene73     Gene74     Gene75     Gene76     Gene77     Gene78 
0.51107715 0.79001510 0.93161980 0.07872220 0.78266705 0.26732820 
    Gene79     Gene80     Gene81     Gene82     Gene83     Gene84 
0.65938384 0.42489203 0.60132528 0.68918845 0.35570327 0.91222852 
    Gene85     Gene86     Gene87     Gene88     Gene89     Gene90 
0.47045551 0.04231126 0.95090973 0.82014324 0.63875498 0.16610824 
    Gene91     Gene92     Gene93     Gene94     Gene95     Gene96 
0.48751480 0.41407388 0.39411049 0.70843687 0.25992595 0.86890515 
    Gene97     Gene98     Gene99    Gene100 
0.32682756 0.13940263 0.08271279 0.38382134 

$Weight
Exp 1 Exp 2 Exp 3 
    3     6     8 

> ## Stouffer's method (with weighting by sample-size)
> S <- Stouffer.test(result)
> str(S)

List of 3
 $ Upper : Named num [1:100] 0.177 0.124 0.988 0.96 0.345 ...
  ..- attr(*, "names")= chr [1:100] "Gene1" "Gene2" "Gene3" "Gene4" ...
 $ Lower : Named num [1:100] 0.8225 0.8756 0.0124 0.0403 0.6547 ...
  ..- 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"

> S

$Upper
       Gene1        Gene2        Gene3        Gene4        Gene5 
0.1774644331 0.1244157053 0.9875708665 0.9596581043 0.3452956126 
       Gene6        Gene7        Gene8        Gene9       Gene10 
0.8611914191 0.7918519760 0.1583118041 0.7758732759 0.1693564223 
      Gene11       Gene12       Gene13       Gene14       Gene15 
0.3480673804 0.8383211446 0.4705782075 0.3773948430 0.4316691704 
      Gene16       Gene17       Gene18       Gene19       Gene20 
0.8841142301 0.0120777647 0.2503590295 0.9177647603 0.7986945981 
      Gene21       Gene22       Gene23       Gene24       Gene25 
0.6451405299 0.7709705970 0.0054370721 0.3885293821 0.5670555202 
```

```
      Gene26       Gene27       Gene28       Gene29       Gene30 
0.4433052288 0.0006740877 0.5499194129 0.1917025143 0.2817956904 
      Gene31       Gene32       Gene33       Gene34       Gene35 
0.2224751857 0.5975953387 0.5354203264 0.7339566197 0.9083610205 
      Gene36       Gene37       Gene38       Gene39       Gene40 
0.4835554623 0.6405639777 0.5589251808 0.8658078854 0.9713295598 
      Gene41       Gene42       Gene43       Gene44       Gene45 
0.3818076326 0.7309998385 0.9802693719 0.8015386523 0.2752314024 
      Gene46       Gene47       Gene48       Gene49       Gene50 
0.7295207329 0.9560794518 0.6933942103 0.2346823580 0.3477517328 
      Gene51       Gene52       Gene53       Gene54       Gene55 
0.5726459992 0.3016146524 0.3319258338 0.4350323222 0.1390692336 
      Gene56       Gene57       Gene58       Gene59       Gene60 
0.7904972953 0.9471882525 0.9764579325 0.1408901253 0.9321998048 
      Gene61       Gene62       Gene63       Gene64       Gene65 
0.7810542702 0.7108324154 0.3172821320 0.2694901059 0.8062078138 
      Gene66       Gene67       Gene68       Gene69       Gene70 
0.3186097575 0.5807557504 0.8729713985 0.3778673459 0.1367814393 
      Gene71       Gene72       Gene73       Gene74       Gene75 
0.1607267354 0.4301867071 0.1219848691 0.0536414435 0.1689083683 
      Gene76       Gene77       Gene78       Gene79       Gene80 
0.8969997032 0.3002118420 0.8568175421 0.4919701637 0.5952828502 
      Gene81       Gene82       Gene83       Gene84       Gene85 
0.0187758289 0.3174655227 0.6487720068 0.1397427957 0.5147735262 
      Gene86       Gene87       Gene88       Gene89       Gene90 
0.9263910377 0.1345751261 0.1760805339 0.5669455902 0.5759484809 
      Gene91       Gene92       Gene93       Gene94       Gene95 
0.4234239901 0.7654857403 0.5457262971 0.3088225491 0.7603225600 
      Gene96       Gene97       Gene98       Gene99      Gene100 
0.1339176158 0.7088096125 0.9074636410 0.9360023011 0.7035028798 

$Lower
     Gene1      Gene2      Gene3      Gene4      Gene5      Gene6 
0.82253557 0.87558429 0.01242913 0.04034190 0.65470439 0.13880858 
     Gene7      Gene8      Gene9     Gene10     Gene11     Gene12 
0.20814802 0.84168820 0.22412672 0.83064358 0.65193262 0.16167886 
    Gene13     Gene14     Gene15     Gene16     Gene17     Gene18 
0.52942179 0.62260516 0.56833083 0.11588577 0.98792224 0.74964097 
    Gene19     Gene20     Gene21     Gene22     Gene23     Gene24 
0.08223524 0.20130540 0.35485947 0.22902940 0.99456293 0.61147062 
    Gene25     Gene26     Gene27     Gene28     Gene29     Gene30 
0.43294448 0.55669477 0.99932591 0.45008059 0.80829749 0.71820431 
    Gene31     Gene32     Gene33     Gene34     Gene35     Gene36 
0.77752481 0.40240466 0.46457967 0.26604338 0.09163898 0.51644454 
    Gene37     Gene38     Gene39     Gene40     Gene41     Gene42 
```

```
0.35943602 0.44107482 0.13419211 0.02867044 0.61819237 0.26900016
    Gene43     Gene44     Gene45     Gene46     Gene47     Gene48
0.01973063 0.19846135 0.72476860 0.27047927 0.04392055 0.30660579
    Gene49     Gene50     Gene51     Gene52     Gene53     Gene54
0.76531764 0.65224827 0.42735400 0.69838535 0.66807417 0.56496768
    Gene55     Gene56     Gene57     Gene58     Gene59     Gene60
0.86093077 0.20950270 0.05281175 0.02354207 0.85910987 0.06780020
    Gene61     Gene62     Gene63     Gene64     Gene65     Gene66
0.21894573 0.28916758 0.68271787 0.73050989 0.19379219 0.68139024
    Gene67     Gene68     Gene69     Gene70     Gene71     Gene72
0.41924425 0.12702860 0.62213265 0.86321856 0.83927326 0.56981329
    Gene73     Gene74     Gene75     Gene76     Gene77     Gene78
0.87801513 0.94635856 0.83109163 0.10300030 0.69978816 0.14318246
    Gene79     Gene80     Gene81     Gene82     Gene83     Gene84
0.50802984 0.40471715 0.98122417 0.68253448 0.35122799 0.86025720
    Gene85     Gene86     Gene87     Gene88     Gene89     Gene90
0.48522647 0.07360896 0.86542487 0.82391947 0.43305441 0.42405152
    Gene91     Gene92     Gene93     Gene94     Gene95     Gene96
0.57657601 0.23451426 0.45427370 0.69117745 0.23967744 0.86608238
    Gene97     Gene98     Gene99    Gene100
0.29119039 0.09253636 0.06399770 0.29649712

$Weight
Exp 1 Exp 2 Exp 3
    3     6     8
```

6 Setup

This vignette was built on:

```
> sessionInfo()
```

```
R version 3.0.1 (2013-05-16)
Platform: x86_64-apple-darwin10.8.0 (64-bit)

locale:
[1] ja_JP.UTF-8/ja_JP.UTF-8/ja_JP.UTF-8/C/ja_JP.UTF-8/ja_JP.UTF-8

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

other attached packages:
[1] metaSeq_0.99.0     snow_0.3-12        NOISeq_2.0.0
[4] Biobase_2.20.1     BiocGenerics_0.6.0
```

```
loaded via a namespace (and not attached):
[1] tools_3.0.1
```

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.