Analyse d'expression différentielle pour un séquençage de l'ARN avec edgeR

-

12 décembre 2025

Table des matières

1 Résumé **1**

2 Introduction **2**

3 Méthodologie **2**

3.1 Introduction 2

3.2 Téléchargement des données 2

3.3 Création d'un objet *DGEList* 2

3.4 Annotation 2

3.5 Filtration 3

3.6 Normalisation 4

4 Résultats et discussions **4**

4.1 Exploration des données 4

4.2 Matrice de design 5

4.3 Dispersion BN des données 6

4.4 Expression differential des gènes 7

4.5 Gene Ontology Analysis 8

5 Conclusion **9**

6 Références **9**

1 Résumé

edgeR peut être utilisé pour l'identification des gènes exprimés différemment. Nous allons procéder à une analyse des données issues d'une expérience de « *paired design RNA-seq* » de tissus tumoraux et non tumoraux (normaux). Nous ferons référence à une étude (1) portant sur des carcinomes épidermoïdes buccaux et des tissus normaux correspondants provenant de trois patients. Nous utiliserons edgeR pour détecter les gènes dont l'expression diffère entre ces deux types de tissus.

2 Introduction

Le séquençage de l'ARN (*RNA-seq*) est une méthode permettant de séquencer un ensemble de molécules d'ARN. Le séquençage d'ARN est utilisé pour déterminer quels segments d'ADN ont été transcrits en ARN et la quantité un gène est exprimé, afin de mieux comprendre la fonction des différents gènes (2).

edgeR est un package R conçu pour l'analyse d'expression différentielle des données de « *RNA-seq count* ». Il peut détecter des différences entre deux groupes ou plus quand au moins un des groupes a effectué des mesures répétées (3).

3 Méthodologie

3.1 Introduction

3.2 Téléchargement des données

Nous commonçons par charger les *libraries* pertinentes.

```
library(org.Hs.eg.db)
library(limma)
library(edgeR)
```

Les données sont téléchargées du **DOI** (*the Digital Object Identifier*). Avant de lire les données, nous avons supprimé les colonnes qui n'étaient pas pertinentes pour notre étude. Nous avons également renommé certaines colonnes (`idRefSeq` à `RefSeqID`, `nameOfGene` à `Symbol`, `numberOfExons` à `NbrOfExons`).

```
rawdata <- read.delim("TableS1.txt", check.names = FALSE, stringsAsFactors = FALSE)
head(rawdata, 3)
```

```
##    RefSeqID    Symbol NbrOfExons   8N 8T   33N 33T   51N  51T
## 1 NM_182502 TMPRSS11B         10 2592  3  7805 321  3372    9
## 2 NM_003280     TNNC1          6 1684  0  1787   7  4894  559
## 3 NM_152381     XIRP2         10 9915 15 10396  48 23309 7181
```

3.3 Création d'un objet *DGEList*

Nous stockerons nos données dans l'objet *DGEList* pour faciliter leur manipulation. Cet objet, conçu spécifiquement pour edgeR, peut être manipulé comme une liste. *DGEList* est un structure de données qui comprend toutes les composantes nécessaires (une matrice *counts* contenant les integer *counts*, un data-frame *samples* contenant des informations sur les échantillons et les librairies, et un data-frame *genes* contenant des annotations sur les gènes) dans un seul objet (4).

```
y <- DGEList(counts = rawdata[,4:9], genes = rawdata[,1:3])
```

3.4 Annotation

L'étude (1) a été réalisée en 2010 ; par conséquent, certains des `RefSeqID`s ne sont plus utilisés. Nous filtrons les *RefSeq ID*s qui ne sont plus disponibles dans l'annotation actuelle du **NCBI** (fournie par le package `org.Hs.eg.db`).

```
idfound <- y$genes$RefSeqID %in% mappedRkeys(org.Hs.egREFSEQ)
y <- y[idfound,]
dim(y) # 15534 sur 15668 RefSeqIDs sont retenus
```

```
## [1] 15533     6
```

*Entrez Gene ID*s sont les identifiants numériques uniques attribués aux gènes par le NCBI (5). Pour chaque gène, *Entrez Gene* donne des détails tels que le nom du gène et sa localisation chromosomique. Nous ajoutons les *Entrez Gene ID*s à l'annotation.

```
egREFSEQ <- toTable(org.Hs.egREFSEQ)
head(egREFSEQ, 3)
```

```
##   gene_id accession
## 1       1 NM_130786
## 2       1 NP_570602
## 3       2 NM_000014
```

```
m <- match(y$genes$RefSeqID, egREFSEQ$accession)
y$genes$EntrezGene <- egREFSEQ$gene_id[m]
```

Ensuite, nous mettons à jour `symbol` en utilisant les *Entrez Gene ID*s.

```
egSYMBOL <- toTable(org.Hs.egSYMBOL)
head(egSYMBOL, 3)
```

```
##   gene_id symbol
## 1       1   A1BG
## 2       2    A2M
## 3       9   NAT1
```

```
m <- match(y$genes$EntrezGene, egSYMBOL$gene_id)
y$genes$Symbol <- egSYMBOL$symbol[m]
head(y$genes, 3)
```

```
##    RefSeqID    Symbol NbrOfExons EntrezGene
## 1 NM_182502 TMPRSS11B         10     132724
## 2 NM_003280     TNNC1          6       7134
## 3 NM_152381     XIRP2         10     129446
```

3.5 Filtration

Plusieurs transcrits *RefSeq* correspondent au même gène. Nous ne gardons qu'un seul transcrit *RefSeq* (celui avec le *count* le plus élevé) pour chaque gène.

```
o <- order(rowSums(y$counts))
y <- y[o,]
d <- duplicated(y$genes$Symbol)
y <- y[!d,]
nrow(y)
```

```
## [1] 10510
```

Tous les transcrits se retrouvent au moins 50 fois dans au moins un des 6 cas, donc ce n'est pas nécessaire de filtrer les gènes qui sont faiblement exprimés.

Ensuite, nous recalculons la taille du `library`.

```
y$samples$lib.size <- colSums(y$counts)
```

Et nous utilisons les *Entrez Gene ID*s comme les noms de lignes.

```
rownames(y$counts) <- rownames(y$genes) <- y$genes$EntrezGene
y$genes$EntrezGene <- NULL
```

3.6 Normalisation

La normalisation permet d'éliminer les effets techniques présents dans les données afin de minimiser l'impact des biais techniques sur les résultats (6,7). Nous utilisons la fonction `normLibSizes()` qui utilise la normalisation *TMM* (*trimmed mean of M-values*) pour tenir compte des différences de composition entre les *libraries*.

```
y <- normLibSizes(y)
y$samples
```

```
##     group lib.size norm.factors
## 8N      1  7397598    1.1542497
## 8T      1  7124442    1.0619357
## 33N     1 15260793    0.6556112
## 33T     1 13651143    0.9484143
## 51N     1 19318441    1.0892960
## 51T     1 14382783    1.2045134
```

4 Résultats et discussions

4.1 Exploration des données

Nous voulons voir s'il existe des valeurs aberrantes ou d'autres corrélations dans notre échantillon. La fonction `plotMDS` produit un plot dans lequel les distances entre les échantillons représentent approximativement les différences d'expression (8).
Sur le plot (**figure 1**), la dimension 1 sépare les échantillons tumoraux des échantillons normaux et la dimension 2 correspond au numéro du patient. On peut observer que les échantillons non tumoraux sont proches les uns des autres, tandis que les échantillons tumoraux sont plus espacés. Donc, les échantillons tumoraux sont plus hétérogènes que les échantillons normaux.

```
plotMDS(y)
```

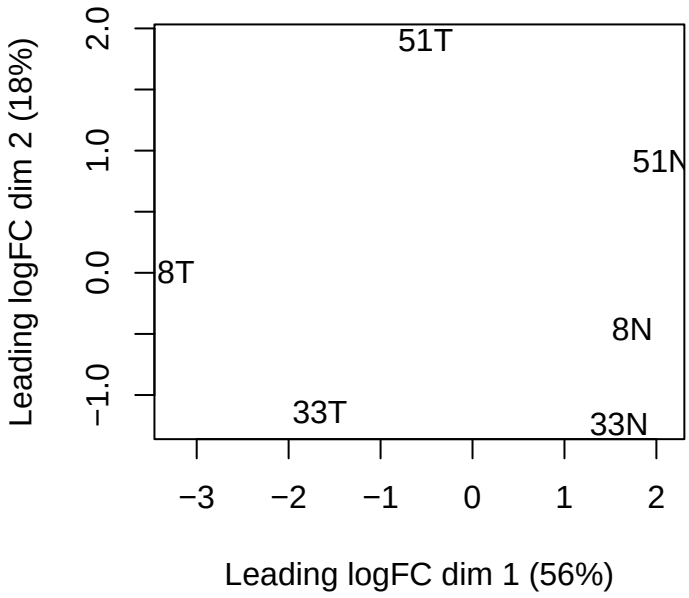


Figure 1: plotMDS(y)

4.2 Matrice de design

Ensuite, nous voulons tester l'expression différentielle entre les tissus tumoraux et les tissus normaux chez les patients.

```
Patient <- factor(c(8, 8, 33, 33, 51, 51))
Tissue <- factor(c("N", "T", "N", "T", "N", "T"))
data.frame(Sample = colnames(y), Patient, Tissue)
```

```
##   Sample Patient Tissue
## 1     8N       8      N
## 2     8T       8      T
## 3    33N      33      N
## 4    33T      33      T
## 5    51N      51      N
## 6    51T      51      T
```

```
design <- model.matrix(~Patient+Tissue)
rownames(design) <- colnames(y)
design
```

```
##     (Intercept) Patient33 Patient51 TissueT
## 8N            1         0         0       0
## 8T            1         0         0       1
## 33N           1         1         0       0
```

```
## 33T           1         1         0       1
## 51N           1         0         1       0
## 51T           1         0         1       1
## attr(,"assign")
## [1] 0 1 1 2
## attr(,"contrasts")
## attr(,"contrasts")$Patient
## [1] "contr.treatment"
##
## attr(,"contrasts")$Tissue
## [1] "contr.treatment"
```

4.3 Dispersion BN des données

Pour comprendre la variabilité biologique, nous estimons la dispersion globale du *dataset*.

```
y <- estimateDisp(y, design, robust = TRUE)
y$common.dispersion
```

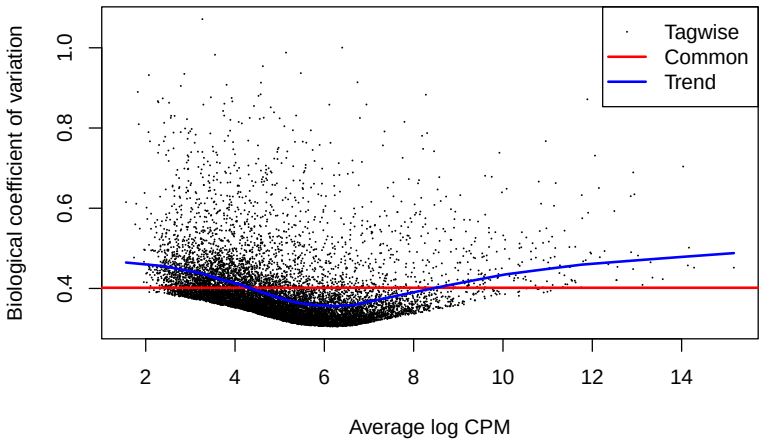
```
## [1] 0.1613756
```

La racine carrée du *common dispersion*, `0.1613`, nous donne le BCV (biological coefficient of variation), `0.402`.

```
sqrt(y$common.dispersion) # BCV
```

```
## [1] 0.4017158
```

```
plotBCV(y)
```



4.4 Expression differential des gènes

Nous utilison la fonction `glmFit` (fits negative binomial generalized log-linear model to each genes read count.)

```
fit <- glmFit(y, design)
```

Ensuite, la fonction `glmLRT` est utilisée pour effecturer des *likelihood ratio tests* pour les différences entre le tissu tumoral et le tissu normal.

```
lrt <- glmLRT(fit)
```

Nous utilisons la fonction `topTags` pour voir les résultats.

```
topTags(lrt)
```

```
## Coefficient:  TissueT
##         RefSeqID   Symbol NbrOfExons     logFC   logCPM        LR       PValue          FDR
## 5737   NM_000959    PTGFR          3 -5.201023 4.822267 100.46226 1.206744e-23 1.268288e-19
## 5744   NM_198966    PTHLH          4  3.881888 5.820335  84.29578 4.260245e-20 2.238759e-16
## 1288   NM_001847   COL4A6         45  3.710900 5.709635  78.26706 9.001057e-19 3.153370e-15
## 10351  NM_007168    ABCA8         38 -3.996432 5.022051  77.53105 1.306522e-18 3.432885e-15
## 5837   NM_005609     PYGM         20 -5.495113 6.075033  74.74014 5.369333e-18 1.128634e-14
## 487    NM_173201   ATP2A1         22 -4.623578 6.040006  73.58113 9.658372e-18 1.691825e-14
## 27179  NM_014440    IL36A          4 -6.178402 5.486198  72.16644 1.977909e-17 2.969689e-14
## 196374 NM_173352    KRT78          9 -4.258876 7.697534  70.84623 3.861808e-17 5.073451e-14
## 6387   NM_199168   CXCL12          3 -3.717669 5.864198  68.91229 1.029414e-16 1.202127e-13
## 83699  NM_031469 SH3BGRL2          4 -3.947822 5.622290  68.36318 1.359935e-16 1.429291e-13
```

```
colnames(design)
```

```
## [1] "(Intercept)" "Patient33"   "Patient51"   "TissueT"
```

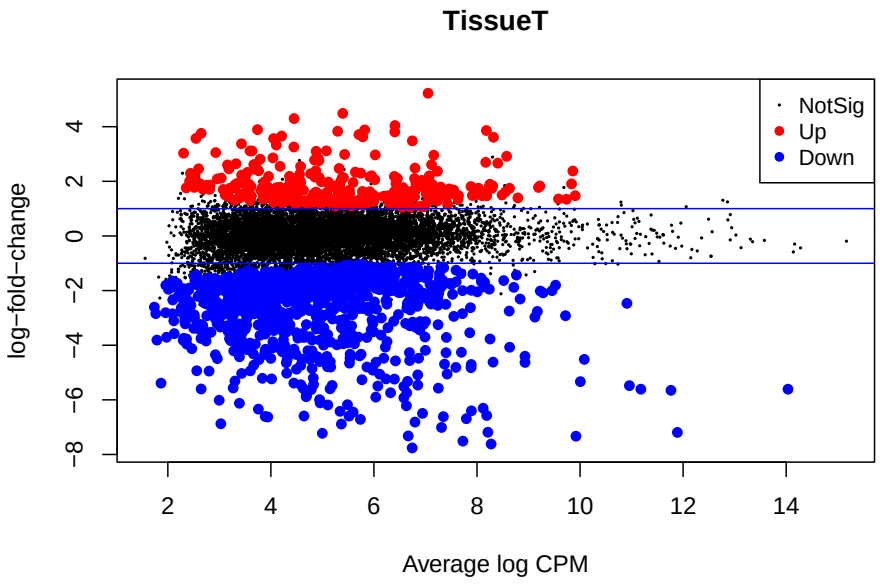
```
o <- order(lrt$table$PValue)
cpm(y)[o[1:10],]
```

```
##                8N          8T       33N        33T        51N          51T
## 5737    53.286956   0.9252284  28.28544   0.926860  82.448258   2.59751429
## 5744     5.387253  78.1157149  10.59455 133.854037   5.940076 104.07373912
## 1288    11.945647 137.0659837   6.19681  96.470682   4.514458  57.26075940
## 10351   56.449039   3.3043873  41.77849   2.239912  83.636273   6.34947937
## 5837   163.842749   2.9078608 126.63482   1.235813 103.167244   5.94542159
## 487    114.537676   3.3043873 155.12015   4.016393 107.634181   9.29332890
## 27179   42.980907   1.3217549 182.30616   3.475725  38.111529   0.05772254
## 196374 399.125153  21.9411314 615.48318  50.436634 153.206446   4.73324826
## 6387    63.827233   3.0400363  68.46476   6.179067 188.514259  17.95170985
## 83699  103.177600   5.4191951 124.03615   5.715637  50.894573   5.65680889
```

```
summary(decideTests(lrt))
```

```
##        TissueT
## Down       946
## NotSig    9243
## Up         321
```

```
plotMD(lrt)
abline(h=c(-1, 1), col="blue")
```



4.5 Gene Ontology Analysis

```
go <- goana(lrt)
topGO(go, ont="BP", sort="Up", n=10, truncate=45)
```

```
##                                          Term Ont    N  Up Down         P.Up       P.Down
## GO:0022008                       neurogenesis  BP 1083  74  120 1.277061e-11 7.917927e-03
## GO:0009888                 tissue development  BP 1294  82  199 3.631218e-11 1.150777e-15
## GO:0007399        nervous system development  BP 1556  92  162 7.160014e-11 2.115949e-02
## GO:0007155                      cell adhesion  BP  945  63  160 1.687833e-09 2.547931e-16
## GO:0048513          animal organ development  BP 1850  99  267 2.914578e-09 1.359572e-17
## GO:0048731                system development  BP 2433 120  309 4.077632e-09 1.454654e-12
## GO:0060429            epithelium development  BP  771  54   96 5.630044e-09 5.315945e-04
## GO:0007275 multicellular organism development  BP 2857 134  336 7.638940e-09 2.317471e-09
## GO:0048699             generation of neurons  BP  913  60  103 7.739626e-09 8.254978e-03
## GO:0030154              cell differentiation  BP 2603 125  324 8.483777e-09 4.414787e-12
```

```
go <- goana(lrt)
topGO(go, ont="BP", sort="Down", n=10, truncate=45)
```

```
##                                          Term Ont    N  Up Down         P.Up       P.Down
```

```
## GO:0003012                muscle system process  BP  278   6  103 8.572576e-01 7.674966e-39
## GO:0006936                   muscle contraction  BP  208   5   84 7.664698e-01 5.304959e-35
## GO:0003008                       system process  BP  963  44  198 4.263559e-03 1.740569e-31
## GO:0055001              muscle cell development  BP  152   3   67 8.480195e-01 7.743519e-31
## GO:0055002     striated muscle cell development  BP  129   3   59 7.592912e-01 2.467449e-28
## GO:0042692          muscle cell differentiation  BP  297   7   89 8.077281e-01 9.232699e-26
## GO:0061061         muscle structure development  BP  485  12  119 8.128848e-01 1.505117e-25
## GO:0051146 striated muscle cell differentiation  BP  218   6   73 6.595613e-01 1.581137e-24
## GO:0030239                   myofibril assembly  BP   60   0   36 1.000000e+00 5.292179e-23
## GO:0032501     multicellular organismal process  BP 3975 163  497 1.102136e-06 3.932672e-22
```

5 Conclusion

6 Références

1. Tuch BB, Laborde RR, Xu X, Gu J, Chung CB, Monighetti CK, et al. Tumor transcriptome sequencing reveals allelic expression imbalances associated with copy number alterations [Internet]. 2010. Available from: https://journals.plos.org/plosone/article?id=10.1371/journal.pone.0009317#abstract0

2. Ian C. Nova PhD. RNA-seq (RNA sequencing) [Internet]. 2025. Available from: https://www.genome.gov/genetics-glossary/RNA-seq

3. Robinson MD, McCarthy DJ, Smyth GK. edgeR: A bioconductor package for differential expression analysis of digital gene expression data. [Internet]. 2009. Available from: https://pmc.ncbi.nlm.nih.gov/articles/PMC2796818/

4. DGEList: DGEList constructor [Internet]. 2025. Available from: https://www.rdocumentation.org/packages/edgeR/versions/3.14.0/topics/DGEList

5. Maglott D, Ostell J, Pruitt KD, Tatusova T. Entrez gene: Gene-centered information at NCBI [Internet]. 2007. Available from: https://pmc.ncbi.nlm.nih.gov/articles/PMC1761442/

6. Robinson MD, Oshlack A. A scaling normalization method for differential expression analysis of RNA-seq data [Internet]. 2010. Available from: https://pmc.ncbi.nlm.nih.gov/articles/PMC2864565/

7. Singh V, Kirtipal N, Song B, Lee S. Normalization of RNA-seq data using adaptive trimmed mean with multi-reference [Internet]. 2024. Available from: https://pmc.ncbi.nlm.nih.gov/articles/PMC11107385/#:~:text=Abstract,characteristic%20curve%20and%20differential%20expression.

8. plotMDS.DGEList: Multidimensional scaling plot of distances between digital gene expression profiles [Internet]. 2025. Available from: https://www.rdocumentation.org/packages/edgeR/versions/3.14.0/topics/plotMDS.DGEList

9. Chen Y, McCarthy D, Baldoni P, Ritchie M, Robinson M, Smyth G. edgeR: Differential analysis of sequence read count data. User's guide [Internet]. 2025. Available from: https://www.bioconductor.org/packages/release/bioc/vignettes/edgeR/inst/doc/edgeRUsersGuide.pdf

10. Dunning M, Pereira B, Rueda O, Santiago ID, Samarajiwa S. Differential expression analysis using edgeR [Internet]. 2015. Available from: https://bioinformatics-core-shared-training.github.io/cruk-bioinf-sschool/Day3/Supplementary-RNAseq-practical.pdf