

Pre-Processing for the Zebrafish RNA-Seq Gene-Level Counts

Davide Risso

Modified: April 13, 2014. Compiled: November 5, 2019.

This vignette describes the pre-processing steps that were followed for the generation of the gene-level read counts contained in the *Bioconductor* package [zebrafishRNASeq](#).

Contents

1	Sample preparation and sequencing	1
2	Read alignment and expression quantitation	1
3	Loading the zebrafish data into <i>R</i>	2
4	Session info	2

1 Sample preparation and sequencing

Olfactory sensory neurons were isolated from three pairs of gallein-treated and control embryonic zebrafish pools and purified by fluorescence activated cell sorting (FACS) [1]. Each RNA sample was enriched in poly(A)+ RNA from 10–30 ng total RNA and 1 μ L (1:1000 dilution) of Ambion ERCC ExFold RNA Spike-in Control Mix 1 was added to 30 ng of total RNA before mRNA isolation. cDNA libraries were prepared according to manufacturer’s protocol. The six libraries were sequenced in two multiplex runs on an Illumina HiSeq2000 sequencer, yielding approximately 50 million 100bp paired-end reads per library.

2 Read alignment and expression quantitation

We made use of a custom reference sequence, defined as the union of the zebrafish reference genome (Zv9, downloaded from Ensembl [2], v. 67) and the ERCC spike-in sequences (<http://tools.invitrogen.com/downloads/ERCC92.fa>). Reads were mapped with TopHat [3] (v. 2.0.4), with the following parameters,

```
--library-type=fr-unstranded -G ensembl.gtf --transcriptome-index=transcript --no-novel-juncs
```

where `ensembl.gtf` is a GTF file containing Ensembl gene annotation.

Gene-level read counts were obtained using the `htseq-count` python script [4] in the “union” mode and Ensembl (v. 67) gene annotation.

Pre-Processing for the Zebrafish RNA-Seq Gene-Level Counts

After verifying that there were no run-specific biases, we used the sums of the counts of the two runs as the expression measures for each library.

3 Loading the zebrafish data into R

To load the gene-level read counts into R, simply type

```
library(zebrafishRNASeq)
data(zfGenes)

head(zfGenes)

##           Ctl1 Ctl3 Ctl5 Trt9 Trt11 Trt13
## ENSDARG00000000001  304 129 339 102   16  617
## ENSDARG00000000002  605 637 406  82  230 1245
## ENSDARG00000000018  391 235 217 554  451  565
## ENSDARG00000000019 2979 4729 7002 7309 9395 3349
## ENSDARG00000000068   89 356  41 149   45  44
## ENSDARG00000000069  312 184  84 269  513  243
```

The ERCC spike-in read counts are in the last rows of the same matrix and can be retrieved in the following way.

```
spikes <- zfGenes[grepl("^ERCC", rownames(zfGenes)),]
head(spikes)

##           Ctl1  Ctl3  Ctl5  Trt9  Trt11  Trt13
## ERCC-00002 97227 38556 68367 148331 169360 100974
## ERCC-00003 10925  6240 11156 36652  21184  21841
## ERCC-00004 379182 179870 256130 679783 529085 311169
## ERCC-00009  2452  1183  1042  1895  3520  1252
## ERCC-00012    0    0    0    0    0    0
## ERCC-00013   89    8    0  205   21    3
```

The typical use of this dataset is the identification of differentially expressed genes between control (Ctl) and treated (Trt) samples. For additional details, exploratory analysis, and normalization of the zebrafish data see [5, 6]. The data are used as a case study for the *Bioconductor* package *RUVSeq*.

4 Session info

```
toLatex(sessionInfo())
```

- R version 3.6.1 (2019-07-05), x86_64-pc-linux-gnu
- Locale: LC_CTYPE=en_US.UTF-8, LC_NUMERIC=C, LC_TIME=en_US.UTF-8, LC_COLLATE=C, LC_MONETARY=en_US.UTF-8, LC_MESSAGES=en_US.UTF-8, LC_PAPER=en_US.UTF-8, LC_NAME=C, LC_ADDRESS=C, LC_TELEPHONE=C, LC_MEASUREMENT=en_US.UTF-8, LC_IDENTIFICATION=C
- Running under: Ubuntu 18.04.3 LTS

Pre-Processing for the Zebrafish RNA-Seq Gene-Level Counts

- Matrix products: default
- BLAS: /home/biocbuild/bbs-3.10-bioc/R/lib/libRblas.so
- LAPACK: /home/biocbuild/bbs-3.10-bioc/R/lib/libRlapack.so
- Base packages: base, datasets, grDevices, graphics, methods, stats, utils
- Other packages: zebrafishRNASeq 1.6.0
- Loaded via a namespace (and not attached): BiocManager 1.30.9, BiocStyle 2.14.0, Rcpp 1.0.2, compiler 3.6.1, digest 0.6.22, evaluate 0.14, highr 0.8, htmltools 0.4.0, knitr 1.25, magrittr 1.5, rlang 0.4.1, rmarkdown 1.16, stringi 1.4.3, stringr 1.4.0, tools 3.6.1, xfun 0.10, yaml 2.2.0

References

- [1] T. Ferreira, S. R. Wilson, Y. G. Choi, D. Risso, S. Dudoit, T. P. Speed, and J. Ngai. Silencing of odorant receptor genes by G Protein $\beta\gamma$ signaling ensures the expression of one odorant receptor per olfactory sensory neuron. *Neuron*, 81:847–859, 2014.
- [2] P. Flicek, M. R. Amode, D. Barrell, K. Beal, S. Brent, D. Carvalho-Silva, P. Clapham, G. Coates, S. Fairley, S. Fitzgerald, et al. Ensembl 2012. *Nucleic Acids Research*, 40(D1):D84–D90, 2012.
- [3] C. Trapnell, L. Pachter, and S. L. Salzberg. TopHat: discovering splice junctions with RNA-Seq. *Bioinformatics*, 25(9):1105–1111, 2009.
- [4] S. Anders, P. T. Pyl, and W. Huber. HTSeq – A Python framework to work with high-throughput sequencing data. *bioRxiv preprint*, 2014. doi:10.1101/002824.
- [5] D. Risso, J. Ngai, T.P. Speed, and S. Dudoit. Using controls for the normalization of RNA-Seq data. *Nature Biotechnology*, 2014. Accepted.
- [6] D. Risso, J. Ngai, T.P. Speed, and S. Dudoit. The role of spike-in standards in the normalization of RNA-seq. In D. Nettleton and S. Datta, editors, *Statistical Analysis of Next Generation Sequence Data*. Springer, 2014.