STANDARD OPERATING PROTOCOL · EnGeL-BIOINF-RNASEQ

RNA sequencing: from reads to differentially expressed genes and gene ontology (FastQC, Trimmomatic, STAR, featureCounts, DESeq2, DAVID)

Protocol ID	EnGeL-BIOINF-RNASEQ
Version	1.1.2
Date	2026-11-20
Status	draft
Area	Bioinformatics (BIOINF)
Authors	Fernanda Benedet de Santo (earthworm analysis and DESeq2 notebook; ORCID 0000-0003-1081-2099); Sarah Louise Cruz (bee analysis, contrasts and gene ontology); Artur Sarmento (bee read processing; ORCID 0000-0003-3736-1531); Luís Cunha (lead; first pipeline notes; review and approval; ORCID 0000-0002-5870-2537)
Reviewed and approved by	Luís Cunha
Laboratory	Environmental Genomics Laboratory (EnGeL-UC), Environmental Ecology & Ecotoxicology Group (E3G), Centre for Functional Ecology, Department of Life Sciences, University of Coimbra · Laboratório Associado TERRA
Contact	luis.cunha@uc.pt · ewormuc@gmail.com
Licence	Text: CC BY 4.0 · Code: MIT
Main hazards	None: computational protocol

1 Purpose and scope

This protocol takes paired-end Illumina RNA-seq reads from organisms **with a reference genome** and produces a table of read counts per gene, a list of differentially expressed genes between conditions, and the gene ontology (GO) and pathway terms enriched among them. It follows total RNA extraction (EnGeL-MOLB-RNA) and nucleic acid QC (EnGeL-MOLB-NANODROP) in the RNA workflow.

The lab's libraries are not strand-specific and are usually sequenced as 2 × 150 bp reads. Settings that depend on this are marked.

For species **without a reference genome** a separate branch is needed (step 8); it is planned and not yet written.

2 Principle

Read quality is checked (FastQC) and adapters and low-quality bases are removed (Trimmomatic). The trimmed reads are aligned to the genome with a splice-aware aligner (STAR), which places reads that span introns. The number of read pairs that fall on each gene is counted from the genome annotation (featureCounts).

DESeq2 models the counts of each gene with a negative binomial distribution. It corrects for differences in sequencing depth between samples (size factors), estimates how much each gene varies between replicates (dispersion), borrowing information across genes because replicates are few, and tests for each gene whether the mean differs between conditions. P-values are adjusted for the thousands of tests made (Benjamini–Hochberg false discovery rate, padj).

The fold changes of genes with few reads are very noisy: a gene with 2 reads in one group and 8 in the other has a fourfold change that means little. Shrinkage (apeglm) pulls such uncertain estimates towards zero and leaves well-supported ones almost unchanged, so the genes at the top of a ranked list, and at the edges of a volcano plot, are those with large *and* reliable changes. Shrinkage changes the fold changes only, not the p-values.

Finally, the lists of genes that go up and down are compared with all genes tested to find GO terms and KEGG pathways that are over-represented (DAVID).

3 Before you start

Item	Details
Input	Paired fastq files (`<sample>_1.fq.gz`, `<sample>_2.fq.gz`) as delivered by the provider
Sample table	CSV with one row per sample: `sample` (as in the file names) and `condition` (e.g. ctl, low, high); add columns for any other factor of the design (e.g. batch, experiment)
Reference	Genome FASTA and an annotation (GTF) made for that same assembly; record source and version
Design	At least 5 biological replicates per condition; 7–8 recommended. The contrasts to test written down before the analysis
Computing	Lab workstation; hands-on and run time [TO CONFIRM]

CRITICAL Check that the annotation belongs to the genome assembly before building the index: chromosome or scaffold names in the GTF must match those in the FASTA, and genes should fall on sensible coordinates (open a few in a genome browser). A public annotation that does not match its assembly gives few or wrong counts; in that case the genome has to be re-annotated and the lab's own annotation used, and recorded.

CRITICAL Write down the design and the contrasts before looking at the results. Choosing contrasts, thresholds or samples after seeing which genes come out changes the meaning of the p-values.

4 Materials

4.1 Computer

- Lab bioinformatics workstation (Linux, many cores and enough memory for the STAR genome index) [TO CONFIRM: minimum memory for the genomes used]

4.2 Consumables

Not applicable.

4.3 Software

- FastQC [TO CONFIRM: version]
- Trimmomatic 0.39 (Java), with its adapter file adapters.fa
- STAR 2.7.11a
- samtools 1.9
- featureCounts, Subread 2.0.6
- R with DESeq2, apeglm, ggplot2, gplots and RColorBrewer (Bioconductor 3.19 in the source notebook) [TO CONFIRM: R and package versions]
- DAVID web server (functional annotation and enrichment) [TO CONFIRM: DAVID version and knowledgebase date at the time of use]
- Reference genome (FASTA) and a gene annotation (GTF) that matches it, for the species studied

5 Reagent preparation

Not applicable. Software installation [TO CONFIRM: installation method on the lab workstation, e.g. conda environment]:

```
if (!require("BiocManager", quietly = TRUE)) install.packages("BiocManager")
BiocManager::install(c("DESeq2", "apeglm"))
install.packages(c("ggplot2", "gplots", "RColorBrewer"))
```

6 Procedure

Set the number of processor threads once (`THREADS=24`, for example) and work in one project folder with sub-folders `0.raw`, `1.qc`, `2.trimmed`, `3.index`, `4.mapping`, `5.counts`, `6.deseq2`, `7.go`. Commands assume that the programs are on the PATH.

6.1 1. Quality check of the raw reads (FastQC)

1. Run FastQC on all raw files and look at the reports (per-base quality, adapter content, overrepresented sequences):

```
fastqc -t $THREADS -o 1.qc/raw 0.raw/*.fq.gz
```

6.2 2. Trimming (Trimmomatic)

2. Trim each pair. Adapters are removed with the lab's adapter file (`ILLUMINACLIP` 2:30:10); then bases below quality 3 at either end, windows of 4 bases with mean quality below 15, and reads shorter than 36 bases:

```
TRIM=/path/to/Trimmomatic-0.39     # folder with trimmomatic-0.39.jar and adapters/adapters.fa
for R1 in 0.raw/*_1.fq.gz; do
  S=$(basename "$R1" _1.fq.gz)
  java -jar $TRIM/trimmomatic-0.39.jar PE -threads $THREADS -phred33 \
    0.raw/${S}_1.fq.gz 0.raw/${S}_2.fq.gz \
    2.trimmed/${S}_1_paired.fq.gz 2.trimmed/${S}_1_unpaired.fq.gz \
    2.trimmed/${S}_2_paired.fq.gz 2.trimmed/${S}_2_unpaired.fq.gz \
    ILLUMINACLIP:$TRIM/adapters/adapters.fa:2:30:10 LEADING:3 TRAILING:3 SLIDINGWINDOW:4:15 MINLEN:36
done
```

3. Run FastQC again on the paired outputs (`fastqc -t $THREADS -o 1.qc/trimmed 2.trimmed/*_paired.fq.gz`) and check that adapter content has gone. Only the `_paired` files are used from here on.

6.3 3. Genome index (STAR)

4. Build the index once per genome and annotation version. `--sjdbOverhang` is the read length minus 1 (149 for 2 × 150 bp reads). `--genomeSAindexNbases` is set from the genome size, as the STAR manual advises: min(14, log2(genome length)/2 − 1), rounded down (about 12 for a 250 Mb genome, 14 for genomes of about 1 Gb or more):

```
GLEN=$(grep -v '^>' genome.fna | tr -d '\n' | wc -c)                    # genome length in bases
SAN=$(awk -v g=$GLEN 'BEGIN{n=int(log(g)/log(2)/2-1); if(n>14) n=14; print n}')
STAR --runThreadN $THREADS --runMode genomeGenerate --genomeDir 3.index \
  --genomeFastaFiles genome.fna --sjdbGTFfile genes.gtf \
  --sjdbOverhang 149 --genomeSAindexNbases $SAN
```

A value lower than the formula only makes mapping a little slower; a value too high for a small genome can make STAR fail or use too much memory.

PAUSE POINT. The index can be kept and reused for every project on the same genome and annotation.

6.4 4. Alignment (STAR)

5. Align each sample and write a coordinate-sorted BAM file directly, named after the sample:

```
for R1 in 2.trimmed/*_1_paired.fq.gz; do
  S=$(basename "$R1" _1_paired.fq.gz)
  STAR --runThreadN $THREADS --genomeDir 3.index \
    --readFilesIn 2.trimmed/${S}_1_paired.fq.gz 2.trimmed/${S}_2_paired.fq.gz \
    --readFilesCommand zcat \
    --outSAMtype BAM SortedByCoordinate --outSAMunmapped Within --outSAMattributes Standard \
    --outFileNamePrefix 4.mapping/${S}_
  samtools index 4.mapping/${S}_Aligned.sortedByCoord.out.bam
done
```

6. For each sample, read `4.mapping/<sample>_Log.final.out` and record the input reads and the percentage of uniquely mapped reads.

> The source notes wrote SAM files and sorted and converted them with samtools afterwards. Writing sorted BAM in STAR gives the same alignments in fewer steps and less disk space. The GTF is already in the index, so it is not given again here.

6.5 5. Counting reads per gene (featureCounts)

7. Count read pairs (fragments) per gene for all samples in one table. The libraries are not strand-specific (`-s 0`):

```
featureCounts -T $THREADS -p --countReadPairs -s 0 -a genes.gtf \
  -o 5.counts/featureCounts_output.txt 4.mapping/*_Aligned.sortedByCoord.out.bam
```

`--countReadPairs` is needed from Subread 2.0.2 onwards (the lab workstation has 2.0.6) for `-p` to count pairs rather than single reads; older versions count pairs with `-p` alone and do not accept the option. Check with `featureCounts -v`.

8. Open `5.counts/featureCounts_output.txt.summary` and record the percentage of assigned reads per sample.

6.6 6. Differential expression (DESeq2)

9. Import the counts and the sample table:

```
library(DESeq2); library(ggplot2); library(gplots); library(RColorBrewer)

fc <- read.table("5.counts/featureCounts_output.txt", header = TRUE, row.names = 1, comment.char = "#",
check.names = FALSE)
counts <- as.matrix(fc[, -(1:5)])                       # drop Chr, Start, End, Strand, Length
colnames(counts) <- sub("_Aligned.sortedByCoord.out.bam$", "", basename(colnames(counts)))

samples <- read.csv("samples.csv", row.names = 1)        # columns: sample (row names), condition, ...
samples$condition <- relevel(factor(samples$condition), ref = "ctl")   # the control is the reference
stopifnot(all(rownames(samples) == colnames(counts)))    # same samples, same order
```

10. Fit the model and check the dispersion plot: the fitted line should follow the cloud of gene estimates, which should fall towards it as mean counts increase:

```
dds <- DESeqDataSetFromMatrix(countData = counts, colData = samples, design = ~ condition)
dds <- DESeq(dds)
plotDispEsts(dds)
```

If the experiment has another factor (for example batch, or two experiments analysed together), include it in the design, e.g. `~ batch + condition`. Analyse separate experiments (e.g. acute and chronic exposures) as separate data sets unless the design was planned to combine them.

11. Transform the counts for plots and look at how the samples group. Use rlog for usual experiments; for large data sets (more than about 30 samples) use `vst(dds, blind = TRUE)` instead, which is much faster and gives practically the same plots. The transformation is used only for plots, never for the test. Look at how the samples group, with a sample-distance heatmap and a principal component analysis:

```
rld <- rlog(dds, blind = TRUE)
d <- as.matrix(dist(t(assay(rld))))
heatmap.2(d, trace = "none", col = colorRampPalette(rev(brewer.pal(9, "Blues")))(100), margins = c(10,
10))
plotPCA(rld, intgroup = "condition")
```

CRITICAL Replicates are never removed only because they look different. A sample is excluded only for a documented laboratory reason (for example a known extraction, library or sample-handling problem, or a sample swap confirmed by other evidence), the decision and reason are recorded, and the analysis with all samples is kept and reported. A sample that groups away from its replicates is a reason to check the laboratory records, not a reason to drop it.

12. List the coefficients and get the results for each planned contrast, with α = 0.05 for independent filtering:

```
resultsNames(dds)                                    # e.g. "condition_high_vs_ctl",
"condition_low_vs_ctl"
res_high <- results(dds, name = "condition_high_vs_ctl", alpha = 0.05)
summary(res_high)
hist(res_high$pvalue, breaks = 50)                   # should be flat with a peak near 0
```

13. Shrink the fold changes (apeglm) for ranking, plots and the fold-change threshold. apeglm works on the coefficients listed by `resultsNames`, i.e. each condition against the reference; for other comparisons (e.g. high vs low) either set that level as the reference and refit, or use `type = "ashr"` with `contrast = c("condition", "high", "low")`:

```
shr_high <- lfcShrink(dds, coef = "condition_high_vs_ctl", type = "apeglm")
plotMA(shr_high, ylim = c(-3, 3))
```

14. Call genes differentially expressed with the lab's criteria, **padj < 0.05 and |log2 fold change| ≥ 0.5**, taking padj from `results()` and the fold change from the shrunken estimates, so that a gene passes only if its change is both large and well supported, and write the tables:

```
out <- data.frame(gene = rownames(res_high), baseMean = res_high$baseMean,
                  log2FC_shrunk = shr_high$log2FoldChange, lfcSE_shrunk = shr_high$lfcSE,
                  log2FC_mle = res_high$log2FoldChange, pvalue = res_high$pvalue, padj = res_high$padj)
out$regulation <- ifelse(!is.na(out$padj) & out$padj < 0.05 & out$log2FC_shrunk >=  0.5, "up",
                  ifelse(!is.na(out$padj) & out$padj < 0.05 & out$log2FC_shrunk <= -0.5, "down", "ns"))
out <- out[order(out$padj), ]
table(out$regulation)
write.csv(out, "6.deseq2/DE_high_vs_ctl.csv", row.names = FALSE)
write.csv(counts(dds, normalized = TRUE), "6.deseq2/normalised_counts.csv")
```

Stricter option Filtering by fold change after the test keeps genes whose change differs from zero and whose estimated change is at least 0.5. The stricter alternative tests against the threshold itself: `results(dds, name = "condition_high_vs_ctl", lfcThreshold = 0.5, alpha = 0.05)`, where the p-value asks whether the change is

larger than 0.5 rather than different from zero. It calls fewer genes, especially with few replicates; use it when a conservative list is needed (for example for a publication) and say which approach was used.

15. Draw a volcano plot from the same table:

```
ggplot(subset(out, !is.na(padj)), aes(log2FC_shrunk, -log10(pvalue), colour = regulation)) +
  geom_point(size = 0.8) +
  geom_vline(xintercept = c(-0.5, 0.5), linetype = 2) +
  scale_colour_manual(values = c(down = "#2166AC", ns = "grey70", up = "#B2182B")) +
  labs(x = "log2 fold change (shrunken)", y = "-log10 p-value") + theme_bw()
ggsave("6.deseq2/volcano_high_vs_ctl.png", width = 6, height = 5, dpi = 300)
```

16. Repeat the last four steps for each planned contrast.
17. If a functional annotation table exists for the genome (gene ID, gene name, description, GO, KEGG), join it to the results by gene ID before writing the final tables, and record its source and version.

PAUSE POINT. Save the DESeq2 object (`saveRDS(dds, "6.deseq2/dds.rds")`) so that plots and contrasts can be redone without refitting.

6.7 7. Gene ontology and pathway enrichment (DAVID)

18. For each contrast, write three gene lists as plain text, one identifier per line: genes up, genes down, and the **background** of all genes tested (genes with a padj value, i.e. not removed by independent filtering):

```
writeLines(out$gene[out$regulation == "up"],   "7.go/high_vs_ctl_up.txt")
writeLines(out$gene[out$regulation == "down"], "7.go/high_vs_ctl_down.txt")
writeLines(out$gene[!is.na(out$padj)],         "7.go/high_vs_ctl_background.txt")
```

19. Check that the identifiers are of a type DAVID recognises for the species (for example NCBI gene IDs or gene symbols). Convert them if needed and record the conversion.
20. In DAVID, upload the up list as a gene list and the background list as background, select the species, and run the functional annotation chart for GO biological process, molecular function and cellular component, and KEGG pathways. Repeat for the down list. Export the charts.
21. Report terms with a Benjamini-adjusted p-value below 0.05. Use both the functional annotation chart (individual terms) and the functional annotation clustering (groups of related, redundant terms, ranked by enrichment score) to see which processes stand out. Record the DAVID version and knowledgebase date.

DAVID only knows species and identifiers in its knowledgebase. For species it does not cover, enrichment needs a GO annotation of the genome itself and an R package such as topGO or clusterProfiler [TO CONFIRM: lab choice].

6.8 8. Branch: species without a reference genome

[TO CONFIRM: protocol to be written.] Without a genome the reads are assembled into a transcriptome (de novo assembly), transcripts are quantified and grouped into genes, and the transcriptome is annotated by similarity to protein databases before the same DESeq2 and enrichment steps (6 and 7). Tools, settings and quality checks for this branch are to be decided by the lab.

7 Calculations

Fold change. DESeq2 reports the base-2 logarithm of the ratio of mean normalised counts between two conditions,

$$\log_2 \mathrm{FC} = \log_2 \frac{\bar{x}_{\mathrm{treatment}}}{\bar{x}_{\mathrm{control}}}$$

where $\bar{x}$ is the mean of the normalised counts of a gene in each condition (dimensionless; normalised counts are reads corrected by the size factor of each sample).

Threshold. |log2FC| ≥ 0.5 means a change of at least $2^{0.5}$ = 1.41-fold, up or down.

Worked example. A gene with mean normalised counts of 200 in the control and 300 in the treatment has log2FC = log2(1.5) = 0.58: above the threshold, so it is called "up" if padj < 0.05. A gene going from 200 to 260 has log2FC = log2(1.3) = 0.38 and is not called, however small its padj.

8 Quality control

Check	Where	Acceptance criterion
Raw and trimmed read quality	FastQC reports (step 1, step 2)	Adapters removed after trimming; [TO CONFIRM: other criteria]
Read pairs kept after trimming	Trimmomatic log	[TO CONFIRM]
Uniquely mapped reads	`<sample>_Log.final.out` (step 4)	≥ 70% (starting value; may be adjusted for the genome and study, with the reason recorded)
Reads assigned to genes	`featureCounts_output.txt.summary` (step 5)	≥ 60% (starting value; may be adjusted, with the reason recorded)
Dispersion fit	`plotDispEsts` (step 6)	Fitted line follows the gene estimates
Replicates group together	Sample-distance heatmap and PCA (step 6)	Any sample that does not is checked against the lab records; exclusion only for a documented laboratory reason
P-value distribution	Histogram (step 6)	Flat with a peak near 0; other shapes are investigated before results are used
Differential expression	Step 6	padj < 0.05 and |log2FC| ≥ 0.5 (shrunken)
Enriched terms	Step 7	Benjamini < 0.05

9 Data and metadata

- **Outputs:** FastQC reports, STAR logs, count table (`featureCounts_output.txt` and `.summary`), DESeq2 object, results table per contrast (shrunken and unshrunken fold changes, p-values, padj, regulation), normalised counts, plots, gene lists and DAVID charts.
- **Where:** project data folder [TO CONFIRM] raw reads deposited in a public archive (ENA or SRA) before publication, with the sample table and the processed count table.

Minimum to record

- Sample table (sample, condition, replicate, other factors) and the planned contrasts
- Sequencing provider, run IDs, read length, library type (not strand-specific)
- Genome assembly and annotation (source, accession, version)
- Software versions (FastQC, Trimmomatic, STAR, samtools, Subread, R, DESeq2, apeglm) and all non-default settings
- Per sample: raw read pairs, pairs after trimming, % uniquely mapped, % assigned to genes
- Thresholds used (padj, |log2FC|, shrinkage method); any excluded sample with the documented reason
- DAVID version and knowledgebase date, identifiers and background used
- MINSEQE information for the archive submission [TO CONFIRM: lab practice]

10 Safety and waste

Not applicable (computational protocol).

11 Troubleshooting

Problem	Likely cause	Solution
Adapter content still high after trimming	Wrong adapter file	Use the lab's `adapters.fa` from the Trimmomatic folder; check the provider's adapter sequences.
STAR stops while building the index	Not enough memory	Lower `--genomeSAindexNbases` or use a machine with more memory; see the STAR manual.
Low unique mapping	Wrong genome or annotation version, contamination, degraded RNA	Check the species and assembly; check RNA quality records (EnGeL-MOLB-NANODROP).
Few reads assigned in featureCounts	GTF from a different assembly, or wrong strand setting	Use the GTF of the same assembly; keep `-s 0` for the lab's unstranded libraries.
`stopifnot` fails when importing	Sample names in the table and the count columns differ or are in a different order	Make the names identical and sort the table to the count columns.
`lfcShrink` error with `coef`	The comparison is not a coefficient of the model	Use a name from `resultsNames(dds)`, set a new reference level, or use `type = "ashr"`.
DAVID does not recognise most genes	Identifier type not supported	Convert identifiers; for unsupported species use a GO annotation and topGO or clusterProfiler.

12 Remarks

Shrinkage is recommended by the DESeq2 authors for ranking genes and for MA and volcano plots, because unshrunken fold changes of genes with few reads are inflated by noise. The source notebooks used unshrunken fold changes; both are kept in the results table so earlier analyses can be compared.

rlog and vst do the same job for plots: they put counts on a scale where genes with few reads do not dominate PCA and heatmaps. rlog behaves slightly better with few samples but becomes slow with many; vst is fast and nearly identical above about 30 samples.

Excluding a replicate changes the result and must be justified by laboratory facts, not by the data: one analysis in the sources re-ran the comparison without one control replicate to see its effect, which is useful as a sensitivity check but is reported alongside, not instead of, the analysis with all samples.

The R steps were written for this protocol; the source notebooks were adapted from a public DESeq2 analysis template by Stephen Turner, which is credited here.

13 References

Bolger, A.M., Lohse, M., & Usadel, B. (2014). Trimmomatic: a flexible trimmer for Illumina sequence data. *Bioinformatics*, 30(15), 2114–2120. https://doi.org/10.1093/bioinformatics/btu170

Dobin, A., Davis, C.A., Schlesinger, F., et al. (2013). STAR: ultrafast universal RNA-seq aligner. *Bioinformatics*, 29(1), 15–21. https://doi.org/10.1093/bioinformatics/bts635

Li, H., Handsaker, B., Wysoker, A., et al. (2009). The Sequence Alignment/Map format and SAMtools. *Bioinformatics*, 25(16), 2078–2079. https://doi.org/10.1093/bioinformatics/btp352

Liao, Y., Smyth, G.K., & Shi, W. (2014). featureCounts: an efficient general purpose program for assigning sequence reads to genomic features. *Bioinformatics*, 30(7), 923–930. https://doi.org/10.1093/bioinformatics/btt656

Love, M.I., Huber, W., & Anders, S. (2014). Moderated estimation of fold change and dispersion for RNA-seq data with DESeq2. *Genome Biology*, 15, 550. https://doi.org/10.1186/s13059-014-0550-8

Zhu, A., Ibrahim, J.G., & Love, M.I. (2019). Heavy-tailed prior distributions for sequence count data: removing the noise and preserving large differences. *Bioinformatics*, 35(12), 2084–2092. https://doi.org/10.1093/bioinformatics/bty895

Sherman, B.T., Hao, M., Qiu, J., et al. (2022). DAVID: a web server for functional enrichment analysis and functional annotation of gene lists (2021 update). *Nucleic Acids Research*, 50(W1), W216–W221. https://doi.org/10.1093/nar/gkac194

Turner, S. (2014). RNA-seq analysis with DESeq2 (analysis template). GitHub Gist https://gist.github.com/stephenturner/f60c1934405c127f09a6

Andrews, S. (2010). FastQC: a quality control tool for high throughput sequence data. Babraham Bioinformatics. [TO CONFIRM: URL and version cited]

14 Version history and citation

Version	Date	Change	By
1.0.0	2026-11-20	First EnGeL-UC version, from the lab's RNA-seq notes (Luís Cunha; used by Artur Sarmento for bees and Fernanda Benedet de Santo for Eisenia andrei) and the DESeq2 notebooks of Fernanda Benedet de Santo and Sarah Louise; FastQC check, fold-change shrinkage and enrichment with DAVID added; server paths, project names, data links and gene lists removed.	Luís Cunha
1.1.0	2026-11-20	Values confirmed by Luís Cunha: at least 5 biological replicates per condition (7–8 recommended); starting QC values of 70% uniquely mapped and 60% assigned reads; fold-change threshold on shrunken values, with the lfcThreshold test as the stricter option; DAVID terms at Benjamini < 0.05, chart and clustering; STAR index parameter set per genome; rlog by default, vst for large data sets; reference genome kept general, with a check of the annotation. Author name: Sarah Louise Cruz.	Luís Cunha

Version	Date	Change	By
1.1.1	2026-11-20	Subread version on the lab workstation confirmed (2.0.6); –countReadPairs kept.	Luís Cunha
1.1.2	2026-11-20	Cross-references to the nucleic acid QC protocols, now split into EnGeL-MOLB-NANODROP, EnGeL-MOLB-QUBIT and EnGeL-MOLB-PICOGREEN.	Luís Cunha

How to cite

Santo FBd, Cruz SL, Sarmento A, Cunha L (2026). RNA sequencing: from reads to differentially expressed genes and gene ontology (FastQC, Trimmomatic, STAR, featureCounts, DESeq2, DAVID). EnGeL-UC protocol EnGeL-BIOINF-RNASEQ, version 1.1.2. Zenodo. DOI [TO CONFIRM].

Licence

Text: CC BY 4.0. Code: MIT. Logos are the property of their institutions and are not covered by the licence.

15 Open questions

1. Software versions: FastQC, R, DESeq2, apeglm; installation method on the workstation.
2. QC acceptance criteria still open: read pairs kept after trimming; FastQC criteria beyond adapter removal.
3. Branch without a reference genome (step 8): tools and settings.
4. Sarah Louise Cruz: ORCID, if she has one.
5. Where outputs are kept; archive and MINSEQE practice; minimum memory of the workstation; DOI; FastQC citation.