GPRO SUITE
BY BIOTECHVANA

RNASeq

User Tutorial

Contents

1. PRELIMINARY INFORMATION

1.1 – Tutorial objective

The objective of this tutorial is to provide the end-user with the necessary training materials and information to manage server-side pipelines and workflows for differential expression analysis (DE) and enrichment in RNASeq and GPRO server side. The tutorial also provides a guideline to familiarize users with the two protocols and two execution modes of RNASeq as described below:

Protocols:

- **"Tophat/Hisat2 & Cufflinks"** is recommended for DE studies when the reference genome has an annotation GTF/GFF file.

- **"Mapping & Counting"** is recommended for DE studies that do not have an associated GTF/GFF file.

Execution modes:

- **Step-by-step mode:** the protocol is executed as a workflow of independent steps (e.g. quality analysis, preprocessing, mapping, transcriptome assembly and/or quantification, differential expression, and enrichment) with each step shown in an separate tab. A scroll down bar is also provided for each step to summarizing the available command line interface (CLI) software for each step.

- **Pipeline mode**: all steps of the protocol are run automatically as a pipeline (i.e. one after another).

For more details on the two protocols and the two execution modes, visit the manual for RNASeq at https://gpro.biotechvana.com/tool/RNAseq/manual/overview

1.2 - Tutorial material and case study

Within this tutorial, we use data from a case study of comparative transcriptomics based on the species *Sparus aurata* that was previously published in Pérez-Sánchez et al. (2019). The tutorial material consists of nine RNAseq samples from spleen biopsies from specimens of *S. aurata.* Specimens were separated into two groups: control (BC) (n = 4) and parasite-infected fishes (BI) (n = 5). In **Table 1,** we provide the nine fastq files with the following SRA Accessions, a summarization of each group and the assignation of samples per group.

Table 1: Samples and case study groups

SRA accession	Library Names	Tags
SRR8255070	ZEC_17.12_02_26232_S7_R1_001.fastq	BC1
SRR8255063	ZEC_17.12_06_26226_S10_R1_001.fastq	BC2
SRR8255062	ZEC_17.12_09_26229_S13_R1_001.fastq	BC3
SRR8255040	ZEC_17.12_13_26243_S16_R1_001.fastq	BC4
SRR8255045	ZEC_17.12_16_26246_S2_R1_001.fastq	BI1
SRR8255041	ZEC_17.12_20_26250_S6_R1_001.fastq	BI2
SRR8255056	ZEC_17.12_24_26254_S10_R1_001.fastq	BI3
SRR8255053	ZEC_17.12_28_26258_S14_R1_001.fastq	BI4
SRR8255030	ZEC_17.12_32_26262_S18_R1_001.fastq	BI5

***BC = control; BI = Infected fish.**

The 9 fastq files can be downloaded from NCBI at https://www.ncbi.nlm.nih.gov/bioproject/PRJNA507368. If you need to help downloading this material from NCBI, contact us for support at https://forum.biotechvana.com.

RefSeq material: To complete the tutorial you need the following reference sequences:

- The genome assembly draft of *S. aurata* (fSpaAur1.1 TorreLaSal release) will be used as a reference genome sequence in the Tophat/Hisat2 & Cufflinks protocol.
- The GTF file associated with the coding genes of the fSpaAur1.1 release will be used as a reference genome sequence in the Tophat/Hisat2 & Cufflinks protocol.
- The RefSeq file for transcripts of *S. aurata* (fSpaAur1.1 release) will be used as a transcriptome reference sequence in the Mapping & counting protocol.
- A .csv file with the functional descriptions and annotations for all gene features of *S. aurata* (fSpaAur1.1 release). This will be used to integrate functional information such as gene ontology (GO categories) descriptions or formal annotations to the results of differential expression.

You can download the RefSeq material from TorreLaSal CSIC Nutrigroup at https://nutrigroup-iats.org/welcome/request_file. For more information, contact Professor Jaume Perez-Sanchez (jaime.perez.sanchez@csic.es).

Alternatively, you can also use the Refseq release provided by NBCI at https://www.ncbi.nlm.nih.gov/assembly/GCF_900880675.1. However, please note that NCBI release for *S.aurat*a differ in size and annotations to the TorreLaSal release and so differential expression results would likely vary from the results presented in this tutorial (which is based on the TorreLaSal release.

GOSeq input material: The tutorial demonstrates how to execute GOseq analyses for DEE using either : "Tophat/Hisat2 & Cufflinks" or "Mapping & Counting". Enrichment analysis are performed with the software GOseq (Young et al., 2010). As *S. aurata* is a customized species for this software, you need 4 input files for the analysis; 1) assayed genes; 2) differential expressed genes; 3) gene sizes, and 4) GO terms per gene.

To facilitate this tutorial, we provide you with the following material:

1. "Assayed genes" → `assayed_genes.txt`
2. "Differentially expressed Genes"→ `diff_genes.csv`
3. "Gene size" → `length_genes.txt`
4. "Go terms" → `Go_final_saurata.txt`

The contents of these four files will differ if the analysis is performed via the "Tophat/Hisat2 & Cufflinks" or "Mapping & Counting" protocols. Nevertheless, the procedure is identical in both cases. For this reason, we provide you these four files pre-created for the "Tophat/Hisat2 & Cufflinks" simply to show you the format of each input file. You can download these files at https://ecampus.biotechvana.com/course/view.php?id=17&section=3.

Remember that these files are only valid for GOseq analyses performed in "Tophat/Hisat2 & Cufflinks". If you want to complete a GOseq analysis under the "Mapping & Counting" protocol, you need to prepare these four files yourself. Similarly, the pre-prepared files for GOSeq analyses will not be valid if you use the NCBI release and so they must be prepared seperately.

1.3 - Experiment design and support

The DE study will be performed by comparing the BI group (Infected fish) with the BC group (the control) using associated fastq files as biological replicates.

If you have any questions, contact us for support at https://forum.biotechvana.com. Alternatively, you can visit our chatbot for immediate online support at https://gpro.biotechvana.com/genie.

1.4 - Installing and activating RNAseq and the Server-Side

RNASeq is a Client Side + Server Side application. You can download the latest version of RNASeq (the client side) at https://gpro.biotechvana.com/download/RNAseq. You can install RNASeq on your PC following the instructions in the manual available at https://gpro.biotechvana.com/tool/rnaseq/manual. Java 11 must be previously installed on your PC to run RNASeq.

The GPRO Server Side can be installed on a PC or remote server as a Cloud Computing resource. However, due to the complexity of installing the program, we distribute the GPRO Server Side in a Docker container (Merkel 2014). The Docker container can be installed easily by following the steps described here: https://gpro.biotechvana.com/tool/gpro-server/manual

Once the GPRO server-side docker has been installed, it must be linked to VariantSeq. To do this, go to [Preferences → Pipeline connection settings] in the top menu and type the following into the configuration Dialog (Fig.1):

1. Your email address: to receive notifications from the server.
2. Host / IP address: here you should type localhost (see Fig.1)
3. Port: This field should only be filled if you run the server-side manually. The default number will be 22.

4. Username and password: Your ID credentials are provided to access the host server.

As shown in Fig.1, you can also check the option “Run GPRO server locally using Docker” to automatically start the GPRO container every time you run RNASeq (if you have this option checked, you do not need to type in the port number). You can test if the app is connected to the Server Side by clicking on the tab “Test connection settings”. Alternatively, if you have installed the Server Side manually (without the Docker), Add the IP of the remote server where the Server Side is hosted and the port information (by default 22). You should also leave the Option “Run GPRO server locally using Docker” unchecked.

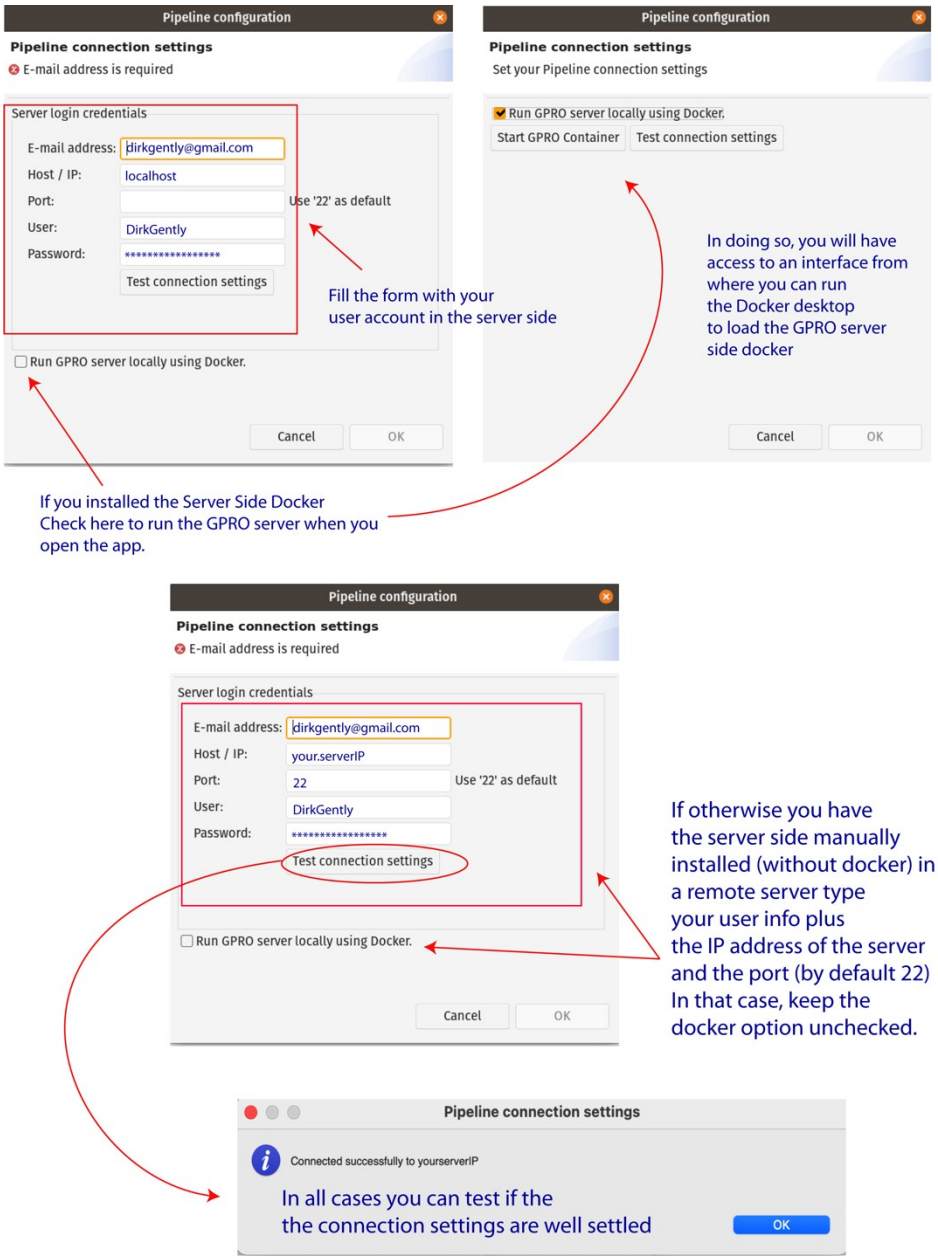

Figure 1. RNASeq connection settings and dialog

2. STEP-BY-STEP MODE TUTORIAL

This mode provides a protocol to complete the DE study in six steps performed as part of step-by-step:

1. Quality analysis
2. Preprocessing
3. Mapping
4. Postprocessing
5. Differential expression

6. GO enrichment

Step-by-Step mode permits two different protocol paths: “Tophat/Hisat2 & Cufflinks” and “Mapping & Counting”.

2.1. - TOPHAT/HISAT2 & CUFFLINKS

“Tophat/Hisat2 & Cufflinks” is recommended for RNA-seq studies when the reference genome has an annotation GTF/GFF file. To complete the tutorial for this protocol, you will need the four fastq files for the infected group (samples from BI1 to BI5), the four fastq files for the control group (samples from BC1 to BC4), and the fasta file with the fSpaAur1.1 reference genome and the associated GTF file.

2.1.1 - Preparing your experiment

When you have already downloaded the tutorial material, open RNASeq and set a directory folder to where you want the aforementioned material on your PC (e.g., your desktop). The space left of the directory browser is the FTP browser for RNAseq. This connects to the directory folder on your PC with your user account on the local host site of the server side. Right click in the FTP browser and create a folder named Tophat_cufflinks. Next, enter this folder and create the following subfolders:

00_raw_data: to deposit the fastq files (BI1-BI5 and BC1-BC4) of both groups. If the fastq files are compressed, you must unzip them
01_quality_analysis: to deposit the results of the quality analysis

02_preprocessed_reads: to deposit the results of the preprocessing analysis

03_refseq: to deposit the fSpaAur1.1 reference genome and its associated GTF

04_ mapping: to deposit the results from the mapping step

05_transcriptome assembly: to deposit the results of the transcriptome assembly and quantification

06_differential_expression: to deposit the results of the differential expression analysis.

07_ go_enrichment_analysis: to deposit for the GO enrichment results from GOseq analysis.

Next, use the FTP browser to move the 9 fastq files from your PC directory browser to the folder 00_raw_data in your local host account on the server side. Then, use the FTP browser to move the fSpaAur1.1 reference genome, the GTF to folder 03_refseq, and the data sets needed for GO enrichment to folder 07_ go_enrichment_analysis. The whole process is shown in Video 1.

Video 1. Setting a directory folder and organizing the user's account before starting

2.1.2 - Quality analysis and preprocessing

To proceed with quality analysis and processing, go RNASeq Protocols → Step-by-Step Mode → Tophat/Hisat & Cufflinks Protocol. A submenu will appear in the workspace showing the following tabs: Preprocessing, Mapping, Transcriptome Assembly, Differential Expression test, and GOseq**.** Each tab references to a step of the DE analysis protocol with its reference genome and annotation file correspondent.

- Quality analysis

For quality analysis, we use FASTQC (Andrews, 2016). To access the interface of FASTQ in RNASeq, go to the step-by-step menu path Tophat/Hisat2 & Cufflinks → Preprocessing → Quality Analysis → FASTQC and proceed as shown in Video 2.

Video 2. Performing a quality analysis with FASTQC.

A FastQC report is an HTML file (Figure 2) containing the sections detailed below and that can be used to check the quality of a sample:

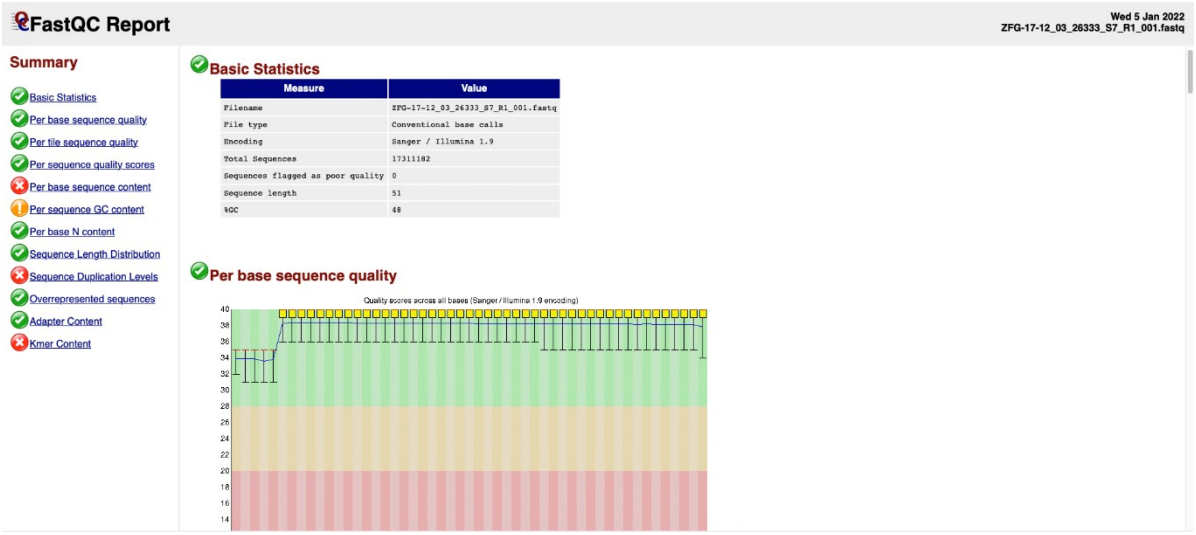

Figure 2. FASTQC Report example.

- **Basic Statistics:** general information on the input FASTQ file such as sample name, type of quality score encoding, the total number of reads, read length, and GC content.
- **Per base sequence quality:** a box-and-whisker plot showing aggregated quality score statistics for every position along all reads in the file.
- **Per tile sequence quality:** this graph will only appear in your results if you are using an Illumina library that retains its original sequence identifiers. The graph

allows you to look at the quality scores from each tile across all bases to see if there was a loss in the quality at only one part of the flowcell.

- **Per sequence quality scores**: a plot of the total number of reads vs. the average quality score over the full length of that read
- **Per base sequence content:** this plot reports the percent of bases called for each of the four nucleotides at each position across all reads in the file.
- **Per sequence GC content**: plot of the number of reads vs. GC% per read. The displayed Theoretical Distribution assumes a uniform GC content for all reads.
- **Per base N content**: percent of bases at each position or bin with no base call, i.e., 'N'.
- **Sequence Length Distribution:** plot showing the distribution of fragment sizes.
- **Sequence Duplication Levels**: percentage of reads of a specific sequence in the file that are present a given number of times in the file.
- **Overrepresented sequences**: list of sequences that appear more than expected in the file. Only the first 50bp are analyzed. A sequence is considered overrepresented if it accounts for ≥ 0.1% of the total reads. Each overrepresented sequence is compared to a list of common contaminants to help with identification.
- **Adapter Content:** cumulative plot of the fraction of reads where the sequence library adapter sequence is identified at the indicated base position. Only adapters specific to the library type are searched.
- **Kmer Content:** measures the count of each short nucleotide of length k (default = 7) starting at each position along the read. Any given Kmer should be evenly represented across the length of the read.

Expected results

When FASTQC is finished, you will obtain a report for each fastq file in your output folder.

The expected results are available at **Quality Analysis**

For more details on the FASTQC report, visit **https://www.bioinformatics.babraham.ac.uk/projects/fastqc/**

You can check the status of the quality analysis in the main menu under Pipelines Jobs → Job Tracking System and clicking on the tracking panel (Figure 3). By right clicking on the panel, you can update, clear, or delete a process. You can also see a log file of the process to check if something failed or which commands were used in the analysis. Finally, you can even rerun an analysis directly from the tracking panel.

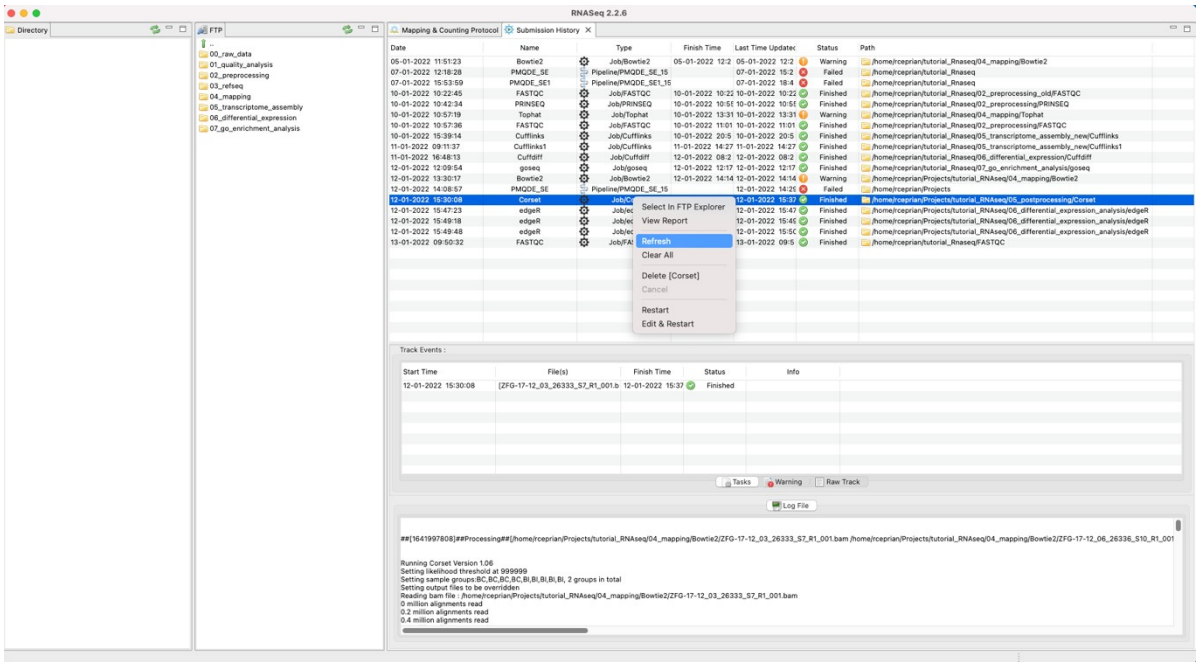

Figure 3. Tracking panel to see the status of each job executed by RNAseq on the server side.

- Preprocessing

Once the quality analysis has finished, the next step of the protocol is to preprocess the fastq files (from BI1-BI5 and BC1-BC4) with low quality sequences.

- Filter samples with PRINSEQ

To apply filters and clean your fastq files (from BI1-BI5 and BC1-BC4), you can use PRINSEQ (Schmieder and Edwards, 2011). You can filter all samples by quality, size, and Ns content. To do this, go to the Step-by-step menu path Tophat/Hisat2 & Cufflinks → Preprocessing → Trimming & Cleaning → PRINSEQ and follow the instructions in Video 3.

Video 3. Filter samples by quality, size, and Ns content with PRINSEQ.

Optional: you can repeat the FASTQC quality analysis to check if PRINSEQ removed all sequencing artifacts and if necessary, execute PRINSEQ a second time.

Expected results from the PRINSEQ preprocessing analysis

When PRINSEQ is complete, you will receive your fastq libraries free of adapters in your output folder.

The expected results are available at the following link **PRINSEQ**

Remember you can check if the job was successfully completed by accessing the job tracking panel of RNASeq

To learn more about PRINSEQ, see **http://prinseq.sourceforge.net**

NOTES: The exact methods used for preprocessing depends on personal preference: in this case, we use PRINSEQ, but you would have also to use TRIMMOMATIC (Bolger et al., 2014), CUTADAPT (Martin 2011) and other parameters.

2.1.3 - Mapping

Mapping aligns the reads of each fastq library to the respective regions on the reference genome where the reads likely originated. Mapping the reads to the reference genome typically involves the alignment of millions of short reads to the genome using algorithms for fast alignment implemented using mapper tools.

To complete the mapping step via Tophat/Hisat2 & Cufflinks with the GTF annotation file, we map the preprocessed fastq files (from BI1-BI5 and BC1-BC4) on the fSpaAur1.1 reference genome. The Step-by-Step menu offers you three mappers TopHat (Kim et al., 2013; Trapnell et al., 2012), Hisat2 (Kim et al., 2015), and STAR (Dobin et al., 2013). In this tutorial, we will use TopHat. To start, select the Step-by-Step menu path, Tophat/Hisat2 & Cufflinks → Mapping → Tophat, and proceed as indicated in Video 4.

Video 4. Mapping fastq libraries on the fSpaAur1.1 reference genome using Tophat and the GTF annotation file.

Expected results from mapping analysis

When TOPHAT is completed, you will receive a bam file for each sample with the reads mapped against the reference genome.

The expected results are available at the following link **Mapping**

You can check the job status by accessing the job tracking panel. Pay particular attention to the log file metrics showing the % of reads successfully mapped. An acceptable value is over 80% of reads mapped per fastq library. If the % is lower than 70%, try preprocessing the samples again to improve cleaning of the fastq libraries.

To learn more about TOPHAT see, **https://ccb.jhu.edu/software/tophat/index.shtml**

2.1.4 - Transcriptome Assembly

The next step of this protocol is the transcriptome assembly step performed by the tool Cufflinks of Cufflinks package (Trapnell et al., 2012). This step is optional but useful for obtaining information about transcript isoforms. Cufflinks assembles the alignments into a parsimonious set of transcripts and, taking into account biases in library preparation protocol, estimates the relative abundances of these transcripts. The interface for Cufflinks also runs cuffcompare, cuffmerge, and cuffquant in the background to respectively compare your transcripts to known transcripts provided the GTF. It also merge (if you wish) the assemblies obtained per fastq library to obtain a consensus transcriptome and this way quantify the expression patterns in FPKMs. To perform the transcriptome assembly, go to the Step-by-Step menu path, Tophat/Hisat2 & Cufflinks → Transcriptome assembly → Cufflinks and proceed as indicated in Video 5.

Video 5. Transcriptome assembly was performed with Cufflinks using the bam files from the mapping steps and the fSpaAur1.1 genome and its associated GTF file.

Expected results from transcriptome assembly analysis:

When Cufflinks is complete, you will receive a bam file for each sample with the reads mapped against the reference genome.

The expected results are available at the following link **Cufflinks**

Remember you can check if the job was successfully completed by accessing the job tracking panel of RNASeq

To learn more about Cufflinks package, see **http://cole-trapnell-lab.github.io/cufflinks/manual/**

2.1.5 - Differential Expression analysis

You can perform the differential expression analysis with the tools Cuffdiff tool Cufflinks (Trapnell et al., 2012), which consists of taking the normalized read count data and performing statistical analysis to discover quantitative changes and differences in expression levels between two or more experimental groups. The interface of Cufflinks also runs Cuffnorm and CummbeRbund (Goff et al., 2019) in the background to normalize the expression levels of the transcripts and obtain some statistics from each test.

We will perform a differential expression comparison between the BI group (Infected fish) versus the BC group (the control). To do this, go to the Step-by-Step menu path, Tophat/Hisat2 & Cufflinks → Differential Expression Analysis → Cuffdiff and follow the instructions in Video 6.

Video 6. Differential expression analysis with Cuffdiff.

Once the differential expression tests have finished, you can add annotations (GOs, descriptions, protein IDs, enzyme codes, etc) to the result files using the .csv file with the functional descriptions and annotations for all gene features of *S. aurata* (you can get it by clicking here).

Expected results from the differential expression analysis:

When Cuffdiff is complete, you will receive the results of differential expression sample with the reads mapped against the reference genome.

The expected results are available at the following link **Cuffdiff**

Bear in mind that Cuffdiff reports the differential expression analysis at distinct levels, genes, isoforms, CDS, TSS, promoters and you will have a full report for each analysis. Significant results have an FDR-adjusted p-value (q-value) < 0.05.

Remember you can check if the job was successfully completed by accessing the job tracking panel of RNASeq

To leaen more about Cufflinks package, see **http://cole-trapnell-lab.github.io/cufflinks/manual/**

For the next step, you will need another GPRO application named Worksheet that can be downloaded from this link https://gpro.biotechvana.com/download/Worksheet

Video 7. Adding annotations to the result file of differential expression with Worksheet.
In Video 7, we provide you with instruction to install and use Worksheet to add annotations to the result file of differential expression from the annotation file already created of the *S. aurata* genome.

For the Worksheet manual, visit https://gpro.biotechvana.com/tool/worksheet/manual

2.1.6 - GOSeq

The next step of this protocol is a GOSeq analysis to determine the enrichment of Gene Ontology features using the software GOSeq (Young et al., 2010). Note that this should also be done for any other annotation line metabolic pathways, SignalP domains, etc. For simplicity, we will only focus on differential expression results at the gene level. The GOSeq analysis can be performed using reference data from native or customized species. Native species in GOSeq analysis have gene lengths and gene categories automatically because they are stored in the GOSeq local database. Thus, the user only needs to provide the assayed genes and differentially expressed genes files. In contrast, for customized species in GOSeq analysis, the user must provide gene lengths and gene categories because they are not in the native GOSeq database. Also, the user must provide the assayed genes and differentially expressed genes files.

As *S. aurata* is a customized species for GOSeq, you must prepare four files individual files for the assayed genes, differential expressed genes, gene sizes, and GO terms per genes. If you used the fSpaAur1.1 reference genome, you may use the prepared files (See Section 1.2). If you used the NCBI genome release, you must prepare these four files yourself. When these files are ready, move them from your directory browser to the folder 07_go_enrichment_analysis of your server-side user account using the FTP browser.

When you are ready to perform the GO enrichment, go to the Step-by-Step menu path, Tophat/Hisat2 & Cufflinks → GOseq → GOseq, and follow this instructions in Video 8.

Video 8. GO enrichment analysis with GOseq from results obtained from differential expression analyses with Cufflinks.

Expected results from GOSeq analysis:

When GOSeq is complete, you will receive the results of the differential expression sample with the reads mapped against the reference genome.

The expected results are available at the following link **GOSeq**

Remember you can check if the job was successfully completed by accessing the job tracking panel of RNASeq

To learn more about Goseq and its outputs see **https://bioconductor.org/packages/release/bioc/html/goseq.html.**

2.2. - MAPPING & COUNTING PROTOCOL

The current protocol (Mapping & Counting) differs from the Tophat/Hisat2 & Cufflinks protocol in that transcripts are counted to quantify the expression patterns and both the mapping, and the differential expression analysis, are performed without a GTF/GFF file. To complete the tutorial via the mapping & counting protocol, we use the 9 fastq files (BI1 to BI5 and BC1 to BC4) as well as the RefSeq fasta file with the transcripts of the fSpaAur1.1 release. In this this tutorial we will use a transcriptome data set as reference instead of the genome because we are not going to use the GTF file.

2.2.1 - Preparing your experiment

From the previous analyses, you should have the following; 1) the RNASeq app connected with the server side, 2) the data and material provided in the 1.2 section, 3) your directory folder selected (and accessible via the directory browser), and 4) access to your user account via the FTP browser of RNASeq.

When you are ready, go to your user account in the local host and create a folder named “Mapping_counting”. Inside this folder you should then create the following subfolders:

02_preprocessed_reads: to deposit the results of the preprocessing analysis.

03_refseq: to deposit the fSpaAur1.1 reference transcriptome.

04_ mapping: to deposit the results from the mapping step.

05_postprocessing: to deposit the results from transcriptome clustering and quantification.

06_differential_expression: to deposit the results of the differential expression analysis.

07_ go_enrichment_analysis: to deposit the GO enrichment results from GOseq analysis.

You do not need to create the folders “00_raw_data” and “01_quality_analysis” because the preprocessing steps are common for both protocols “Tophat/Hisat2 & Cufflinks” and “Mapping & counting”. See the next Section 2.2.2 for more details.

Next, open RNAseq and go to the Step-by-Step menu path RNASeq Protocols → Step-by-Step Mode → Mapping & Counting Protocol. A new submenu will appear in the workspace organizing the different steps that are required to perform the analysis (Preprocessing, Mapping, Postprocessing, Diff. Expression, GOseq).

2.2.2 - Quality analysis and preprocessing

The tools and analyses for data preprocessing are the same as those used during the Tophat/Hisat2 & Cufflinks protocol in Section 2.1.2 . This means that you do not need to repeat these analyses, just do the following:

- Move the fastq files preprocessed with the Tophat/Hisat2 & Cufflinks protocol from the Tophat_cufflinks/02_preprocessing folder to the Mapping_counting/02_preprocessing folder.
- Move the fasta file with the fSpaAur1.1 reference transcriptome from your directory browser to the Mapping_counting/03_refseq. The GTF file is not necessary.

2.2.3 - Mapping

The Mapping and counting protocol provides you with three mappers:Bowtie2 (Langmead and Salzberg, 2012), BWA (Li and Durbin, 2009) and Hisat2 (which also lets you perform the analysis without GTF).
In this tutorial, we use Bowtie2. To start, go to the Step-by-Step menu path, Mapping & Counting → Mapping → Bowtie2, and proceed as indicated in Video 9.

Video 8. Mapping step using Bowtie2

Expected results from Mapping analysis:

When Bowtie2 is complete, you will receive a bam file per sample with the reads mapped against the reference transcriptome data set.

The expected results are available at the following link **Mapping**

Remember you can check the status of the job via the job tracking panel. Pay particular focus to the log file metrics showing the % or reads successfully mapped. An acceptable

2.2.4 - Postprocessing

The next step of this protocol is postprocessing where you can cluster the reads into overlapping fragments and then count the reads with which each cluster was constituted. To this end, you can use Corset (Davidson and Oshlack, 2014) or HTSeq (Anders et al., 2015). For this tutorial, we use Corset. To perform the clustering and counting of reads with Corset, go to the Step-By-Step menu path Mapping & Counting → Postprocessing → Corset and proceed as indicated in Video 10.

Video 9. Classifying and quantifying the expression patterns with Corset.

Expected results from Postprocessing analysis:

When Corset is complete, you will receive two output files:

1. counts.txt: contains read counts per cluster.
2. clusters.txt: contains the correspondence between clusters and genes.

These files are available at the following link **Corset**

Remember you can check if the job was successfully completed by accessing the job tracking panel of RNASeq

To learn more about Corset, see **https://genomebiology.biomedcentral.com/articles/10.1186/s13059-014-0410-6**

2.2.5 - Differential Expression analysis

The final step is to perform the differential expression analysis within the Mapping & Counting protocol. Again, the comparison will be performed between the BI group (Infected fish) versus the BC group (the control) using the 9 fastq files. To do this, the Step-by-Step mode provides you with two alternative tools for differential expression under the Mapping and Counting protocol. The first one is DESeq (Love et al., 2014), which estimates variance-mean dependences in sequencing read count data). The second one is EdgeR (Robinson et al., 2010), which uses an array of statistical methods based on negative binomial distributions including empirical Bayes estimation, exact tests, generalized linear models, and quasi-likelihood tests. In this tutorial, we use EdgeR. To start, go to the Step-by-Step menu path, Mapping & Counting → Differential Expression Analysis → EdgeR, and do as indicated in Video 11.

Expected results from differential expression analysis:

When EdgeR is complete, you will receive the results of differential expression sample with the reads mapped against the reference genome.

The expected results are available at the following link **edgeR**

Bear in mind that under the edgeR analysis significant results have FDR-adjusted p-values < 0.05.

Remember you can check if the job was successfully completed by accessing the job tracking panel of RNASeq

To learn more about EdgeR, see **https://bioconductor.org/packages/release/bioc/html/edgeR.html**

Under this protocol, you can also perform an enrichment analysis; however, as the tool (GOSeq) and the procedure are the same as those previously indicated in Section 2.1.6 we will not reproduce this analysis here. Nevertheless, if you want to do it by yourself, you can prepare the four input files as indicated in video 8 but using instead the results from the EdgeR analysis.

3. PIPELINE MODE TUTORIAL

The pipeline mode of RNAseq allows you to execute all steps of a protocol automatically as a pipeline. The only exception is the GO enrichment step as this is not yet implemented in the pipeline mode. Thus, the pipeline mode currently finishes after the differential expression analysis. To perform the tutorial in pipeline mode, click on the tab “RNASeq protocols” on the Top menu and then select the option Pipeline Mode. A first graphical list of available pipelines will appear based on distinct combinations of tools (Figure 4). Besides, you can select several filters to list only the appropriate pipelines for an analysis according to different criteria including types of reads, protocol mode, etc. You only need to apply your filters and click run. In doing so, you will pass to the next section, which is a set of nested interfaces to:

1. Upload any input data or material needed by the pipeline (i.e., fastq files, reference genome, or GTF/GFF file).
2. Declare the output folder to deposit the output results.
3. Declare the experiment design (groups to compare, samples belonging to each group, replicates, etc).
4. Configure parameters and options for the tools used at each step (for example, “FastQC” for quality analysis, “Trimmomatic” for preprocessing, “Tophat” for mapping, “Cufflinks” for transcriptome assembly and “Cuffdiff” for differential gene expression).
5. Run the pipeline

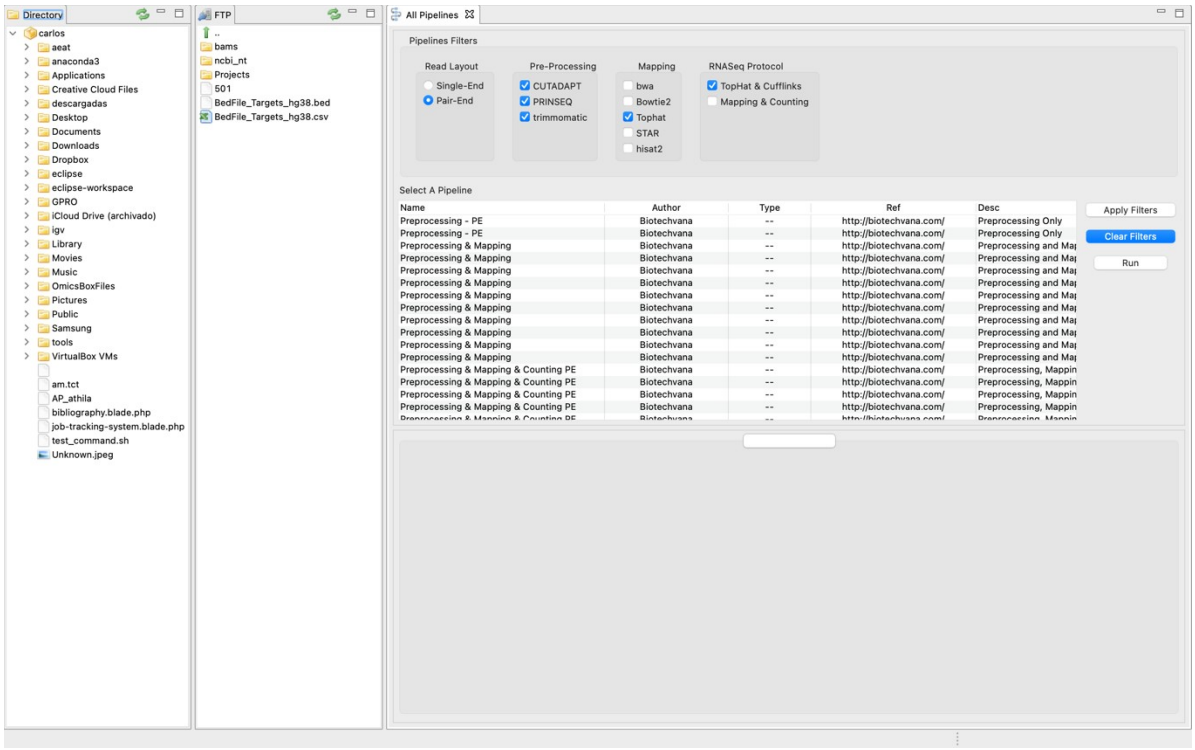

Figure 4. Pipeline designer interface of the pipeline mode

In this part of the tutorial, we reproduce both protocols (“TopHat/Hisat2 & Cufflinks” and “Mapping & Counting”) using the pipeline mode except for the step for the GOSeq analysis that is not currently included in the pipeline mode.

3.1 - TopHat/Hisat2 & Cufflinks protocol

To complete this protocol in pipeline mode, you will need the following material:

- The 5 fastq files for the infected group (samples from BI1 to BI5),
- The 4 fastq files for the control group (samples from BC1 to BC4)
- The fasta file with the fSpaAur1.1 reference genome and its associated GTF file.

When you have this material ready, you can access the pipeline designer interface (RNAseq Protocols → Pipeline mode) and proceed as indicated in video 12.

Video 11. Tophat/Hisat2 & Cufflinks protocol for differential expression analysis executed using pipeline mode of RNASeq.

Expected results from the Tophat/Hisat2 & Cufflinks protocol using the Pipeline execution mode

When the pipeline mode is complete, you will receive the results of differential expression sample with the reads mapped against the reference genome.

The expected results are available at the following link **Pipeline mode: Tophat/Hisat2 & Cufflinks protocol**

Remember you can check if the job was successfully completed by accessing the job tracking panel of RNASeq

To learn more about any tool used in the pipeline and their outputs see their respective manuals, referred in previous sections of this tutorial.

3.2 - Mapping & Counting protocol executed as a pipeline

To complete this protocol in pipeline mode, you will need the following material:

- The 5 fastq files for the infected group (samples from BI1 to BI5).
- The 4 fastq files for the control group (samples from BC1 to BC4).
- The fasta file with the fSpaAur1.1 reference genome.

When you have this material ready, you can access the pipeline designer interface (RNAseq Protocols → Pipeline mode) and proceed as indicated in video 13.

Video 12. Mapping & counting protocol for differential expression analysis executed using the pipeline mode of RNASeq.

Expected results from the Mapping & Counting protocol using the Pipeline execution mode

When the pipeline mode is completed, you will receive the results of differential expression sample with the reads mapped against the reference genome.

The expected results are available in the following link **Pipeline mode: Mapping & Counting protocol**

Remember you can check if the job was successfully completed by accessing the job tracking panel of RNASeq

To learn more about any tool used in the pipeline and their outputs see their respective manuals, referred in previous sections of this tutorial.

4. BIBLIOGRAPHY

- Anders, S., Pyl, P.T. and Huber, W. HTSeq--a Python framework to work with high-throughput sequencing data. *Bioinformatics* 2015;31(2):166-169.

- FastQC: a quality control tool for high throughput sequence data. Bioinformatics 2015;31(2):166-169.

- Bolger, A.M., Lohse, M. and Usadel, B. Trimmomatic: a flexible trimmer for Illumina sequence data. *Bioinformatics* 2014;30(15):2114-2120.

- Davidson, N.M. and Oshlack, A. Corset: enabling differential gene expression analysis for de novo assembled transcriptomes. *Genome Biol* 2014;15(7):410.

- Dobin, A.*, et al.* STAR: ultrafast universal RNA-seq aligner. *Bioinformatics* 2013;29(1):15-21.

- Goff, L., Trapnell, C. and Kelley, D. 2019. CummeRbund: Analysis, exploration, manipulation, and visualization of Cufflinks high-throughput sequencing data

- Kim, D., Langmead, B. and Salzberg, S.L. HISAT: a fast spliced aligner with low memory requirements. *Nature methods* 2015;12(4):357-360.

- Kim, D.*, et al.* TopHat2: accurate alignment of transcriptomes in the presence of insertions, deletions and gene fusions. *Genome Biol.* 2013;14(4):R36.

- Langmead, B. and Salzberg, S.L. Fast gapped-read alignment with Bowtie 2. *Nature methods* 2012;9(4):357-359.

- Li, H. and Durbin, R. Fast and accurate short read alignment with Burrows-Wheeler transform. *Bioinformatics* 2009;25(14):1754-1760.

- Love, M.I., Huber, W. and Anders, S. Moderated estimation of fold change and dispersion for RNA-seq data with DESeq2. *Genome Biol* 2014;15(12):550.

- Martin, M. Cutadapt removes adapter sequences from high-throughput sequencing reads. *EMBnet.journal* 2011;Vol 17(1).

- Merkel, D. Docker: lightweight Linux containers for consistent development and deployment. *Linux Journal* 2014;2014:2.

- Pérez-Sánchez, J.*, et al.* Genome Sequencing and Transcriptome Analysis Reveal Recent Species-Specific Gene Duplications in the Plastic Gilthead Sea Bream (Sparus aurata). *Frontiers in Marine Science* 2019;6(760).

- Robinson, M.D., McCarthy, D.J. and Smyth, G.K. edgeR: a Bioconductor package for differential expression analysis of digital gene expression data. *Bioinformatics* 2010;26(1):139-140.

- Schmieder, R. and Edwards, R. Quality control and preprocessing of metagenomic datasets. *Bioinformatics* 2011;27(6):863-864.

- Trapnell, C.*, et al.* Differential gene and transcript expression analysis of RNA-seq experiments with TopHat and Cufflinks. *Nat.Protoc.* 2012;7(3):562-578.

- Young, M.D.*, et al.* Gene ontology analysis for RNA-seq: accounting for selection bias. *Genome Biol.* 2010;11(2):R14.