Title

A Computational Workflow for Analysis of 3′ Tag-Seq Data

Permalink
https://escholarship.org/uc/item/45j3c779

Journal
Current Protocols, 3(2)

ISSN
2691-1299

Authors
Paropkari, Akshay D
Bapat, Priyanka S
Sindi, Suzanne S
et al.

Publication Date
2023-02-01

DOI
10.1002/cpz1.664

Peer reviewed

Published in final edited form as:

Curr Protoc. 2023 February ; 3(2): e664. doi:10.1002/cpz1.664.

A computational workflow for the analysis of 3’ Tag-Seq data

Akshay D. Paropkari[1,2], **Priyanka S. Bapat**[1,2], **Suzanne S. Sindi**[3,*], **Clarissa J. Nobile**[2,4,*]

[1]Quantitative and Systems Biology Graduate Program, University of California, Merced, CA, USA

[2]Department of Molecular and Cell Biology, School of Natural Sciences, University of California, Merced, CA, USA

[3]Department of Applied Mathematics, School of Natural Sciences, University of California, Merced, CA, USA

[4]Health Science Research Institute, University of California, Merced, CA, USA

Abstract

RNA-sequencing (RNA-seq) is a gold standard method to profile genome-wide changes in gene expression. RNA-seq uses high-throughput sequencing technology to quantify the amount of RNA in a biological sample. With the increasing popularity of RNA-seq, many variations on the protocol have been proposed to extract unique and relevant information from biological samples. 3’ Tag-Seq (also called TagSeq, 3′ Tag-RNA-Seq, and Quant-Seq 3′ mRNA-Seq) is one RNA-seq variation, where the 3’ end of the transcript is selected and amplified to yield one copy of cDNA from each transcript in the biological sample. We present a simple, easy to use, and publicly available computational workflow to analyze 3’ Tag-Seq data. The workflow begins by trimming sequence adapters from raw FASTQ files. The trimmed sequence reads are checked for quality using FastQC, aligned to the reference genome, and read counts are obtained using STAR. Differential gene expression analysis is performed using DESeq2, based on differential analysis of gene count data. The outputs of this workflow are MA plots, tables of differentially expressed genes, and UpSet plots. This protocol is intended for users specifically interested in analyzing 3’ Tag-Seq data. As such, transcript length-based normalizations are not performed within the workflow. Future updates to this workflow could include custom analyses based on the gene counts table as well as data visualization enhancements.

Keywords

3’ Tag-Seq; RNA sequencing; differential gene expression; computational pipeline

[*] To whom correspondence should be addressed, Suzanne Sindi: ssindi@ucmerced.edu, 209-228-4224, Clarissa Nobile: cnobile@ucmerced.edu, 209-303-5468.

CONFLICT OF INTEREST STATEMENT:
Clarissa J. Nobile is a cofounder of BioSynesis, Inc., a company developing diagnostics and therapeutics for biofilm infections. BioSynesis, Inc. was not involved in the content of this manuscript.

INTRODUCTION

RNA sequencing (RNA-seq) is a widely used method to detect genome-wide changes in gene expression (Wang, Gerstein, & Snyder, 2009). It was first used in *Saccharomyces cerevisiae* to identify the gene expression patterns of all genes, exons, and their boundaries across the yeast genome (Nagalakshmi et al., 2008). For humans, as well as many model organisms such as yeast, fruit flies, and mice, RNA-seq has been instrumental in providing high resolution, functionally relevant, genome annotations (Cherry et al., 2012; Gnerre et al., 2011; International Human Genome Sequencing Consortium, 2004; Matthews et al., 2015). Briefly, the RNA-seq protocol begins with the isolation of total RNA, which is typically enriched/selected for polyadenylated (poly(A)) RNA or alternatively depleted for ribosomal RNA (rRNA) (Zhao, Zhang, Gamini, Zhang, & von Schack, 2018). After this step, double stranded complementary DNA (cDNA) is synthesized via reverse transcription from the RNA, resulting in cDNA libraries. The cDNA molecules are fragmented, and sequence adapters are added to the cDNA fragments. Then, the cDNA fragments are subject to high-throughput sequencing to generate short sequence reads, which are used for downstream analyses.

Whole transcript RNA-seq workflows produce data providing information on quantifying gene expression, novel transcripts, alternatively spliced genes, and allele specific expression (Wang et al., 2009). Although this information is valuable, oftentimes, the goal of many biological research projects is to simply identify changes in gene expression patterns between conditions of interest. Given this simplified goal, classical RNA-seq protocols are overly complex and more costly than is necessary towards obtaining genome-wide gene expression changes (Ma et al., 2019; Wang et al., 2009). To simplify classical RNA-seq protocols, recent advances in the field have provided alternative RNA-seq methods that are used today to address specific biological questions (Moll, Ante, Seitz, & Reda, 2014; Morrissy et al., 2011). One of these alternative methods is 3’ Tag-Seq (also called TagSeq, 3′ Tag-RNA-Seq, and Quant-Seq 3′ mRNA-Seq). In this method, cDNA libraries are reverse transcribed only from the 3’ end of the mRNA, resulting in a single copy of cDNA arising from each transcript (Ma et al., 2019; Moll et al., 2014; Torres, Metta, Ottenwälder, & Schlötterer, 2008). Compared to classical RNA-seq methods, 3’ Tag-Seq is simpler, quicker to perform, and less costly while providing sufficient sequencing depth for differential gene expression analysis (Ma et al., 2019). These benefits make 3’ Tag-Seq an ideal choice for researchers whose goal is to identify changes in patterns of gene expression between two or more conditions.

Analysis of sequencing datasets from classical RNA-seq data was complex when this technology was first introduced (Wang et al., 2009). Even today, analysis of RNA-seq data can be challenging because it is highly dependent on the experimental design used to create the sequencing libraries (Conesa et al., 2016). Consequently, there is no “one size fits all” workflow for the analysis of RNA-seq output reads. Here, we present a simple, easy to use, and publicly available computational workflow to analyze 3’ Tag-Seq data, where the user can use default analysis parameters or can adjust parameters to fit their unique experimental design considerations. The workflow begins by trimming sequence adapters from raw FASTQ files (Cock, Fields, Goto, Heuer, & Rice, 2010; Conesa et al., 2016).

The trimmed sequence reads are checked for quality using FastQC, aligned to the reference genome, and read counts are obtained using STAR (Dobin et al., 2013). Differential gene expression analysis is then performed using DESeq2, based on differential analysis of gene count data (Love, Huber, & Anders, 2014). The outputs of this workflow are MA plots, tables of differentially expressed genes, and UpSet plots. This workflow is well-suited for users with minimal computational expertise.

BASIC PROTOCOL 1: Running the 3’ Tag-Seq workflow

In the following section, we describe in detail our 3’ Tag-Seq analysis workflow. Figure 1 provides a summary of the computational workflow. Briefly the pipeline begins with the processing of raw RNA-seq FASTQ files and ends with a table output of differential gene expression (Love et al., 2014).

Necessary Resources:

Hardware—An internet connected machine running Linux 64-bit Ubuntu version 20.04.1 with at least 32 GB RAM. The number of threads can be provided by users, but 16 threads is recommended.

Software—The workflow uses the Conda command line tool environment to install all required software and tools. Conda software can be accessed at https://docs.conda.io/en/latest/miniconda.html. The workflow is saved in a bash script file called `pipeline.sh`. The source code and documentation can be found on GitHub at https://github.com/akshayparopkari/RNAseq/wiki. The Conda environment file is provided in Supplemental file 1.

Other Requirements

- Access to a computational cluster and login information
- Basic knowledge of Unix
- Raw FASTQ sequencing data
- Sample metadata

Downloading the RNA-seq workflow on a local machine

On Linux and MacOS, users can use the built-in Terminal application, and on Windows, users can download and use Git Bash (https://gitforwindows.org/).

1. Navigate to the desired directory to download this folder on your machine.

   ```
   git clone https://github.com/akshayparopkari/RNAseq.git
   ```

   Note: Alternatively, users can click on the green “Code” button on the GitHub page https://github.com/akshayparopkari/RNAseq and click the “Download ZIP” option. Users can then unzip the downloaded folder and save it to a relevant location on their local machines.

2. Make script files executable. [*Copyeditor: I adjusted the step numbers to run consecutively through the protocol. Please ask the authors if this is OK.]

```
cd RNAseq/
chmod u+x pipeline.sh
chmod u+x format_counts_table.py
```

Loading the Conda virtual environment—Conda enables virtual environments that contain the required software packages/libraries to be installed and set up. In this instance, the RNA-seq Conda environment contains the BBMap suite, STAR alignment software, and FASTQC tool (Bushnell, 2014; Dobin et al., 2013). Additionally, required Python and R libraries and their dependencies are also installed.

3. Create Conda environment using Supplemental File 1

```
conda env create -f Supplemental_file_1.yml -n RNAseq
```

Note: Users only need to create the environment once. For subsequent analysis, users can activate the environment to run the analysis using the following command:

```
conda activate RNAseq
```

Creating an input data folder

4. The main script of 3' Tag-Seq is the `pipeline.sh` file. This single bash script contains all the preprocessing steps: QC filtering with BBDuk, generating QC summaries with FastQC, and alignment and gene counting with STAR. The `pipeline.sh` script takes in a single input, which is a folder/directory containing:

    1. all raw FASTQ sequence files AND
    2. the sample metadata Excel file

The raw FASTQ sequence files may either be compressed (using gzip) or uncompressed. The file names must start with the sample ID, followed by the underscore and the rest of the file name. For example, `projectname_date_L001.fastq.gz` should be named `sampleid_projectname_date_L001.fastq.gz`. The first part of the file name before the first underscore dictates the sample the script is processing. The sample metadata file contains all metadata associated with the input samples including sample ID, genotype, condition, treatment, time, etc. For this repository, the sample metadata file must contain at least two columns - *SampleID* and *Condition*. Table 1 is an example of a sample metadata file, where the first two columns *SampleID* and *Condition* are required, and the third column *FASTQ_*file and beyond is optional, but highly recommended. A comprehensive metadata file also enables convenient sample submission to a sequence read archive (SRA), once your manuscript is published.

NOTE: The input directory must contain raw FASTQ files and a sample metadata Excel file. Users may implement a user-defined project structure to organize their RNA-seq data. Please see Cookiecutter Data Science project (https://cookiecutter.readthedocs.io/en/1.7.2/) for ideas on how to best organize computational data.

Transferring data to/from a cloud computing resource to a local machine via command line

5. Below is a common usage of the secure copy `scp` function, which is one of the commands used to transfer files to/from a cloud computing resource. The other command is a secure file transfer protocol `sftp`. Please refer to the cloud computing resource wiki for detailed instructions on `sftp` function.

```
scp FROM TO
```

where FROM is the source location and TO is the destination location.

Third party GUI apps—Users can also use third party clients to transfer files to/from a local computer. FileZilla (https://filezilla-project.org/) for Linux and Windows or Cyberduck (https://cyberduck.io/download) for MacOS and Windows are alternatives to using `scp` or `sftp` to transfer files with drag and drop.

Running the RNA-seq pipeline—Users must activate the RNA-seq Conda environment before attempting to execute the pipeline.

6. Run the RNA-seq pipeline

```
INPUTFOLDER=“path/to/your/input/folder” # enter your data folder
with FASTQ files here
bash pipeline.sh “$INPUTFOLDER” <num_og_threads_to_use> >
“$INPUTFOLDER”/preprocess.log
```

Output files—`pipeline.sh` creates multiple output files, which can be useful to gain insights into specific samples to address any discrepancy in the data. The three important files to check are:

1. `gene_raw_counts.txt`, which is a tab-separated file of raw gene counts for all samples with gene names as the rows and samples as columns

2. `deseq2_lfc.txt`, which is a tab-separated file from the DESeq2 analysis

3. `MA_plot.pdf`, which is a .pdf file depicting volcano plots of log fold changes against mean gene expression

Additional information about other output files:

1. All "`_trimmed.fastq`" ending files are trimmed sequences from BBmap and are saved in the trim_log directory
2. All "`.bam`" ending files are alignment files generated by STAR and are saved in the STAR_log directory
3. All "`ReadsPerGene.out.tab`" ending files are gene count files for each sample generated by STAR and are saved in the `STAR_log` directory

All "`Log.out`", "`Log.final.out`", and "`Log.progress.out`" ending files are intermediary alignment files generated by STAR and are saved in the `STAR_log` directory.

Visualizing overlaps in multiple experimental conditions

7. Use the `overlap_upsetR.R` script to visualize overlaps in genes for multiple experimental conditions. The overlap is represented as an UpSet plot (Lex, Gehlenborg, Strobelt, Vuillemot, & Pfister, 2014). UpSet plots are an extension of Venn diagrams and are useful when there are more than three categories/sets of conditions/samples to consider. The `overlap_upsetR.R` takes one input – either "up" or "down" – to calculate overlap between various samples/conditions. Users need to supply an input directory in the code on line 38, and run the following command to get the output UpSet plot:

# To visualize genes upregulated in multiple conditions/samples

```
overlap_upsetR.R up
```

# To visualize genes downregulated in multiple conditions/samples

```
overlap_upsetR.R down
```

Figure 2 is an example of an UpSet plot output of *Candida albicans* RNA-seq data showing upregulated genes overlapping various conditions.

SUPPORT PROTOCOL 1: Generating genome indices

During the alignment step, STAR utilizes genome index files for mapping sequenced reads to a reference genome. This protocol describes how to generate genome indices for the *Candida albicans* Assembly 21 genome as an example. These steps can be used to generate genome indices for your reference genome of choice.

Necessary Resources:

Hardware—The hardware requirements are the same as for Basic Protocol 1.

Software—The workflow uses the Conda command line tool environment to install all required software and tools. Conda software can be accessed at https://docs.conda.io/en/

latest/miniconda.html. The workflow is saved in a bash script file called `pipeline.sh`. The source code and documentation can be found on GitHub at https://github.com/akshayparopkari/RNAseq/wiki. The Conda environment file is provided in Supplemental file 1.

Other Requirements

1. Access to a computational cluster and login information
2. Basic knowledge of Unix
3. Raw FASTQ sequencing data

1. In your home folder, download *Candida albicans* chromosomal sequences from the *Candida* Genome Database - http://www.candidagenome.org/ (Skrzypek et al., 2017).

```
wget http://www.candidagenome.org/
download/sequence/C_albicans_SC5314/Assembly21/current/
C_albicans_SC5314_A21_current_chromosomes.fasta.gz
gunzip C_albicans_SC5314_A21_current_chromosomes.fasta.gz
```

2. Download the *Candida albicans* genome annotation GTF file from the *Candida* Genome Database.

```
wget http://www.candidagenome.org/download/gff/C_albicans_SC5314/
Assembly21/C_albicans_SC5314_A21_current_features.gtf
gunzip C_albicans_SC5314_A21_current_features.gtf
```

3. Activate the 3’ Tag-Seq Conda environment.

```
module load anaconda3
source activate RNA-seq
```

4. Generate STAR genomes.

```
mkdir ca_genome/
cd ca_genome/
STAR --runMode genomeGenerate --genomeDir ./ --genomeFastaFiles ~/
C_albicans_SC5314_A21_current_chromosomes.fasta
```

5. STAR will generate output index files in the `ca_genome` folder.

GUIDELINES FOR UNDERSTANDING RESULTS

The outputs of this 3-Tag-Seq pipeline are MA plots, tables of differentially expressed genes, and UpSet plots. The output files are saved in the input folder supplied with the script. The DGE output table consists of six numerical columns – baseMean, log2FoldChange, lfcSE, stat, pvalue, and padj. The details of these columns are explained in Analyzing RNA-seq data with DESeq2 - https://www.bioconductor.org/packages/devel/bioc/vignettes/DESeq2/inst/doc/DESeq2.html. To obtain a list of genes that are differentially regulated between the two conditions, users can focus on the log2FoldChange and padj columns. The log2FoldChange indicates the magnitude of change in expression values between the two experimental conditions and the padj value provides statistical significance towards the calculated magnitude change. The availability of these output files including the MA-plot indicates a successful run of this pipeline.

COMMENTARY

Background Information:

We present a simple and user-friendly workflow to analyze 3’ Tag-Seq experimental data. This workflow allows the user to determine the differentially expressed genes in the experimental conditions of interest. This workflow also includes a useful UpSet plot script to allow users to further analyze their differential gene expression data.

Given that the experimental methodology behind 3’ Tag-Seq is focused on the 3’-end of the transcript, information related to splicing events is not captured, and thus is a limitation of this methodology.

We note that this workflow is distinct from other pipelines that analyze traditional RNA-seq experimental data in terms of gene counting. In traditional RNA-seq analyses, gene counts are normalized by the length of the transcript or fragment to obtain comparable count values across genes and across sampling conditions. For traditional RNA-seq experiments, transcripts are fragmented, and reverse transcribed into cDNA; therefore, longer transcripts generate more cDNA, thus necessitating length-based normalization procedures during analysis. For 3’ Tag-Seq experiments, only the 3’-end of the transcript is reverse transcribed, eliminating this need for length-based normalizations during analysis. This advantage of 3’ Tag-Seq facilitates an overall simpler analysis workflow compared to traditional RNA-seq in obtaining accurate read count estimates from biological samples.

Critical Parameters:

The script only takes in a single input – a folder with raw FastQ files. The FastQ files can be either compressed or uncompressed.

Troubleshooting:

See Table 2 for troubleshooting tips.

Supplementary Material

Refer to Web version on PubMed Central for supplementary material.

ACKNOWLEDGEMENTS:

We thank all past and present members of the Nobile and Sindi laboratories for feedback on the manuscript. We are especially grateful to Deepika Gunasekaran for constructive comments and support on the computational workflow as well as to the anonymous reviewers for providing feedback that significantly improved the utility of the workflow. This work was supported by the National Institutes of Health (NIH) National Institute of General Medical Sciences (NIGMS) award number R35GM124594 and by the Kamangar family in the form of an endowed chair to C.J.N. The content is the sole responsibility of the authors and does not represent the views of the funders. The funders had no role in the design of the study; in the collection, analyses, or interpretation of the data; in the writing of the manuscript; or in the decision to publish the results.

DATA AVAILABILITY STATEMENT:

The source files that support the protocol are freely available from the GitHub website (https://github.com/akshayparopkari/RNAseq). The example RNA-seq data used to generate Figure 2 can be obtained from the corresponding authors upon request. The test dataset for the TF028 (*rme1* mutant) strain versus the wildtype strain can be accessed at NCBI GEO (https://www.ncbi.nlm.nih.gov/gds) under accession number GSE200778.

LITERATURE CITED:

Babraham Bioinformatics—FastQC A Quality Control tool for High Throughput Sequence Data. (n.d.). Retrieved November 7, 2021, from https://www.bioinformatics.babraham.ac.uk/projects/fastqc/

Bushnell B (2014). BBMap: A Fast, Accurate, Splice-Aware Aligner (No. LBNL-7065E). Lawrence Berkeley National Lab. (LBNL), Berkeley, CA (United States). Retrieved from https://www.osti.gov/biblio/1241166

Cherry JM, Hong EL, Amundsen C, Balakrishnan R, Binkley G, Chan ET, … Wong ED (2012). Saccharomyces Genome Database: The genomics resource of budding yeast. Nucleic Acids Research, 40(D1), D700–D705. 10.1093/nar/gkr1029 [PubMed: 22110037]

Cock PJA, Fields CJ, Goto N, Heuer ML, & Rice PM (2010). The Sanger FASTQ file format for sequences with quality scores, and the Solexa/Illumina FASTQ variants. Nucleic Acids Research, 38(6), 1767–1771. 10.1093/nar/gkp1137 [PubMed: 20015970]

Conesa A, Madrigal P, Tarazona S, Gomez-Cabrero D, Cervera A, McPherson A, … Mortazavi A (2016). A survey of best practices for RNA-seq data analysis. Genome Biology, 17(1), 1–19. 10.1186/s13059-016-0881-8 [PubMed: 26753840]

Dobin A, Davis CA, Schlesinger F, Drenkow J, Zaleski C, Jha S, … Gingeras TR (2013). STAR: Ultrafast universal RNA-seq aligner. Bioinformatics, 29(1), 15–21. 10.1093/bioinformatics/bts635 [PubMed: 23104886]

Gnerre S, MacCallum I, Przybylski D, Ribeiro FJ, Burton JN, Walker BJ, … Jaffe DB (2011). High-quality draft assemblies of mammalian genomes from massively parallel sequence data. Proceedings of the National Academy of Sciences of the United States of America, 108(4), 1513–1518. 10.1073/pnas.1017351108 [PubMed: 21187386]

International Human Genome Sequencing Consortium. (2004). Finishing the euchromatic sequence of the human genome. Nature, 431(7011), 931–945. 10.1038/nature03001 [PubMed: 15496913]

Lex A, Gehlenborg N, Strobelt H, Vuillemot R, & Pfister H (2014). UpSet: Visualization of Intersecting Sets. IEEE Transactions on Visualization and Computer Graphics, 20(12), 1983–1992. 10.1109/TVCG.2014.2346248 [PubMed: 26356912]

Love MI, Huber W, & Anders S (2014). Moderated estimation of fold change and dispersion for RNA-seq data with DESeq2. Genome Biology, 15(12), 1–21. 10.1186/s13059-014-0550-8

Ma F, Fuqua BK, Hasin Y, Yukhtman C, Vulpe CD, Lusis AJ, & Pellegrini M (2019). A comparison between whole transcript and 3’ RNA sequencing methods using Kapa and Lexogen library preparation methods. BMC Genomics, 20(1), 1–12. 10.1186/s12864-018-5393-3 [PubMed: 30606130]

Matthews BB, Dos Santos G, Crosby MA, Emmert DB, St Pierre SE, Gramates LS, … FlyBase Consortium. (2015). Gene Model Annotations for Drosophila melanogaster: Impact of High-Throughput Data. G3 (Bethesda, Md.), 5(8), 1721–1736. 10.1534/g3.115.018929 [PubMed: 26109357]

Moll P, Ante M, Seitz A, & Reda T (2014). QuantSeq 3′ mRNA sequencing for RNA quantification. Nature Methods, 11(12), i–iii. 10.1038/nmeth.f.376

Morrissy S, Zhao Y, Delaney A, Asano J, Dhalla N, Li I, … Marra M (2011). Tag-Seq: Next-Generation Tag Sequencing for Gene Expression Profiling. In Tag-Based Next Generation Sequencing (pp. 211–241). John Wiley & Sons, Ltd. 10.1002/9783527644582.ch13

Nagalakshmi U, Wang Z, Waern K, Shou C, Raha D, Gerstein M, & Snyder M (2008). The Transcriptional Landscape of the Yeast Genome Defined by RNA Sequencing. Science, 320(5881), 1344–1349. 10.1126/science.1158441 [PubMed: 18451266]

Skrzypek MS, Binkley J, Binkley G, Miyasato SR, Simison M, & Sherlock G (2017). The Candida Genome Database (CGD): Incorporation of Assembly 22, systematic identifiers and visualization of high throughput sequencing data. Nucleic Acids Research, 45(D1), D592–D596. 10.1093/nar/gkw924 [PubMed: 27738138]

Torres TT, Metta M, Ottenwälder B, & Schlötterer C (2008). Gene expression profiling by massively parallel sequencing. Genome Research, 18(1), 172–177. 10.1101/gr.6984908 [PubMed: 18032722]

Wang Z, Gerstein M, & Snyder M (2009). RNA-Seq: A revolutionary tool for transcriptomics. Nature Reviews Genetics, 10(1), 57–63. 10.1038/nrg2484

Zhao S, Zhang Y, Gamini R, Zhang B, & von Schack D (2018). Evaluation of two main RNA-seq approaches for gene quantification in clinical RNA sequencing: PolyA+ selection versus rRNA depletion. Scientific Reports, 8(1), 4781. 10.1038/s41598-018-23226-4 [PubMed: 29556074]

INTERNET RESOURCES:

Candida albicans assembly 21 chromosomal sequence FASTA file - http://www.candidagenome.org/download/sequence/C_albicans_SC5314/Assembly21/current/C_albicans_SC5314_A21_current_chromosomes.fasta.gz

Candida albicans assembly 21 chromosomal features GTF file - http://www.candidagenome.org/download/gff/C_albicans_SC5314/Assembly21/C_albicans_SC5314_A21_current_features.gtf

FileZilla software - https://filezilla-project.org/

CyberDuck software - https://cyberduck.io/download

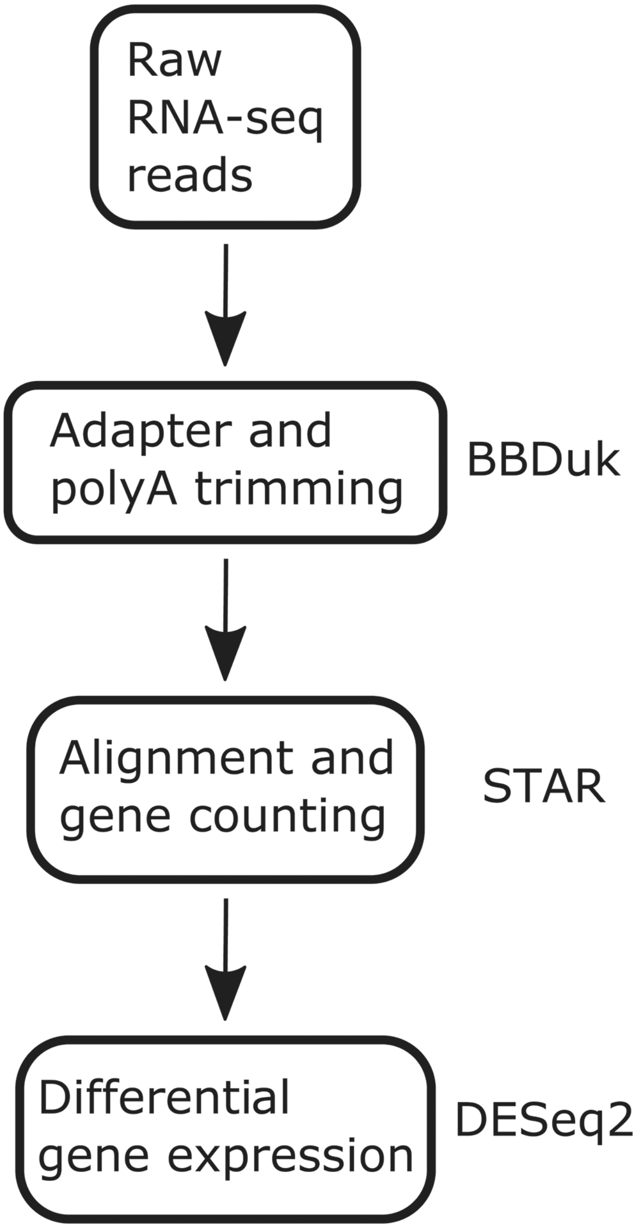
Raw RNA-seq reads
Adapter and polyA trimming
BBDuk
Alignment and gene counting
STAR
Differential gene expression
DESeq2

Figure 1: 3' Tag-Seq analysis workflow.

(i) Adapters added to raw RNA-seq reads are trimmed using BBDuk. (ii) a quality control (QC) report is generated for trimmed reads using FastQC. (iii) Reads passing the QC check are aligned to the reference genome using STAR and a gene count table is created. (iv) The gene count table is used to run differential gene expression analysis using DESeq2, and the DESeq2 output is saved as a table to a file for downstream usage.

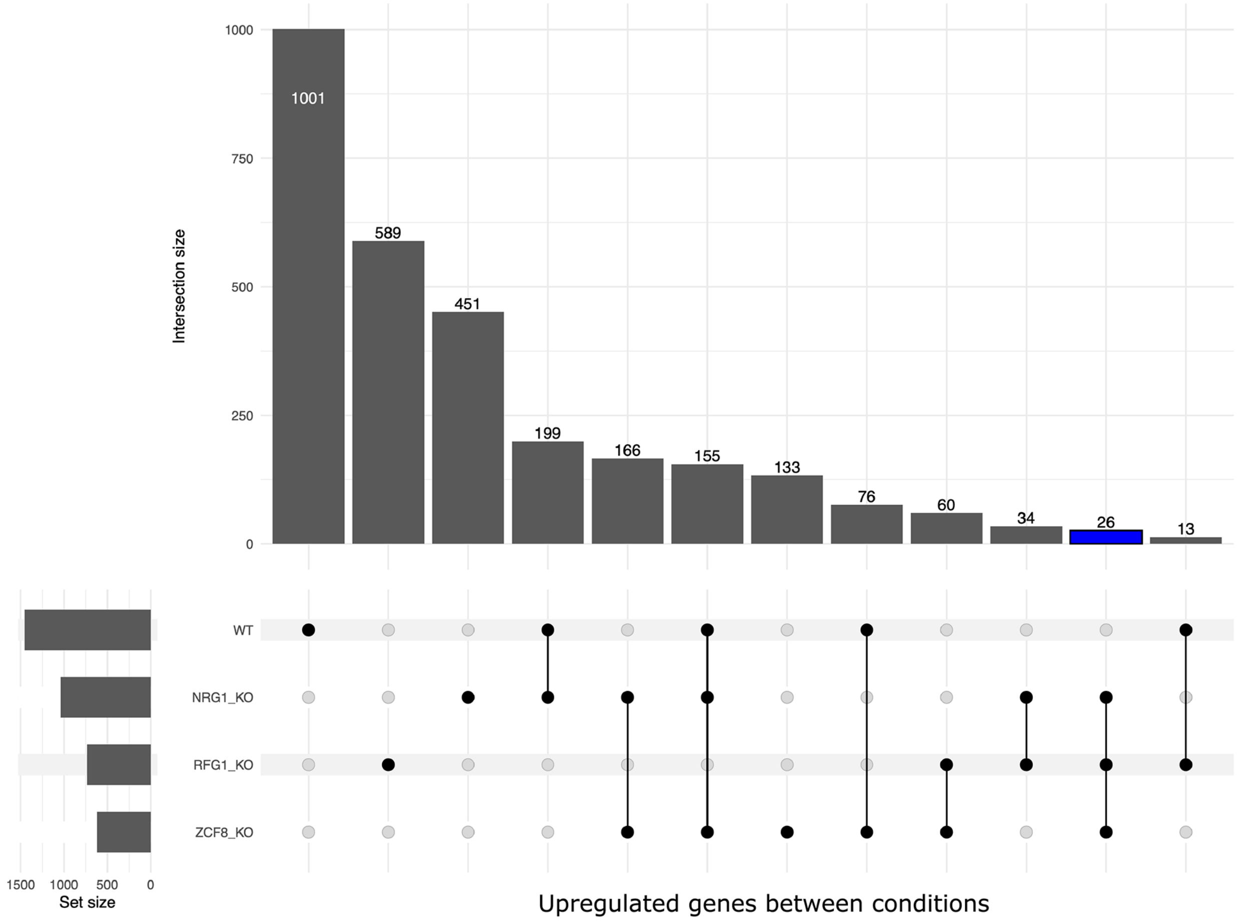

Figure 2: Example UpSet plot output.

UpSet plot output of example *Candida albicans* RNA-seq data for the wildtype (WT) strain and three knockout (KO) mutant strains – NRG1_KO, RFG1_KO and ZCF8_KO. The bar plot on the top represents the overlap of upregulated genes in each condition. The horizontal bar on the left represents the number of significantly upregulated genes in each condition. The black circles and lines at the bottom show the categories for which overlap is calculated as indicated in the bar chart on the top. For example, the fourth bar on the top represents 199 significantly upregulated genes observed in the WT and NRG1-KO conditions. The blue bar highlights the overlap of significantly upregulated genes observed in the three KO conditions.

Table 1:

Sample metadata file. Users can use this file as a template to generate their metadata file.

SampleID	Condition	FASTQ_file	Other_Sample_Info
Sample1A	WT	Sample1A_S8_L001_R1_001.fastq.gz	…
Sample1B	Mutant	Sample1B_S8_L001_R1_001.fastq.gz	…
Sample2A	WT	Sample2A_S8_L001_R1_001.fastq.gz	…
Sample2B	Mutant	Sample2B_S8_L001_R1_001.fastq.gz	…
Sample3A	WT	Sample3A_S8_L001_R1_001.fastq.gz	…
Sample3B	Mutant	Sample3B_S8_L001_R1_001.fastq.gz	…
…	…	…	…

Table 2:

Sources and Solutions to Potential Issues

Issue	Possible Cause	Solution
No gene count table generated by STAR	STAR was unable to locate the reference genome	Follow steps listed in STAR documentation under Section 2 - https://github.com/alexdobin/STAR/blob/master/doc/STARmanual.pdf
No output from DESEq2 analysis	Metadata file was formatted incorrectly	Follow steps listed in GitHub Wiki - https://github.com/akshayparopkari/RNAseq/wiki/4.-Creating-input-data-folder