
UTAP Documentation

Release 1.0.0

Refael Kohen

Mar 07, 2023

Contents

1	UTAP demo website	3
2	Analysis pipeline steps and reports	5
2.1	Analysis pipeline steps	5
2.2	Pipeline report	6
2.3	Annotation file	7
2.4	Examples of reports	7
3	Installation	9
4	User guide	11
4.1	Registration to the system	11
4.2	User datasets	12
4.3	Import Input data	12
4.4	Run analysis	14
4.5	Customization	20
5	Releases	21
6	Source code	23
6.1	UTAP source code	23
6.2	Dependencies	23
7	License	25
8	Author	27
9	Acknowledgments	29

RNA-Seq technology is routinely used to characterize the transcriptome and detect gene expression differences among cell types, genotypes and conditions. Advances in short-read sequencing instruments such as Illumina Next-Seq, have yielded easy-to-operate machines, with higher throughput, at a lower price per base. However, processing this data requires bioinformatics expertise to tailor and execute specific solutions for each type of library preparation.

In order to enable fast and user-friendly data analysis, we developed an intuitive and scalable transcriptome pipeline that executes the full process, starting from sequences (RNA-Seq and bulk MARS-Seq), and ending with sets of differentially expressed genes. Output files are placed in a structured folder system, and summarization of the results is displayed in a rich and comprehensive report containing dozens of plots, tables and links.

CHAPTER 1

UTAP demo website

Navigate to <http://utap-demo.weizmann.ac.il>, and login using:

username: testuser

password: utap1234

Note: This is a demonstration site for UTAP, showcasing the user interface and examples of run results. It provides partial functionality, including the capability to rerun the DESeq step on existing analyses, but does not support running new analyses.

The screenshot shows the UTAP demo website interface. At the top is a dark navigation bar with the text 'UTAP - NGS PIPELINES' and several menu items: 'User Datasets', 'Upload data', 'Run pipeline', and 'Help'. Below the navigation bar is the 'Analyses List' section, which contains a table with two rows of analysis data. Each row includes a 'Delete' button, the analysis name, run status, pipeline name, and creation date. A red box highlights the 'Run Deseq again with other parameters' button for the first analysis, with a red arrow pointing to it from the right.

	Name	Run status	Pipeline	Created	
Delete	20180814_114552_RNA-seq-example	SUCCESSFUL	Transcriptome RNA-seq	Oct. 29, 2018, 4:08 p.m.	Run Deseq again with other parameters
Delete	20180814_102305_MARS-seq-example	SUCCESSFUL	Transcriptome Mars-seq	Oct. 29, 2018, 4:01 p.m.	Run Deseq again with other parameters

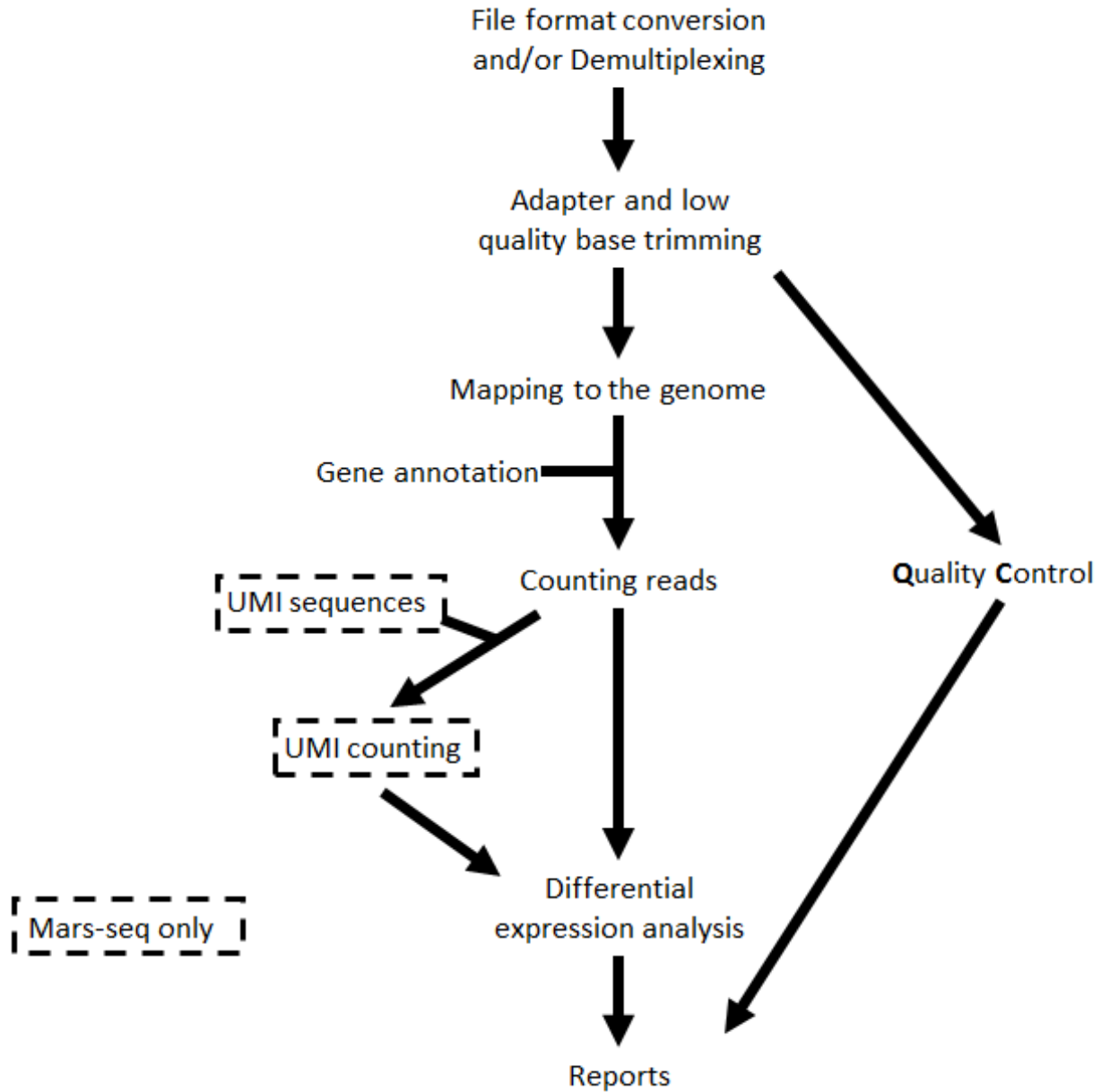
Analysis pipeline steps and reports

2.1 Analysis pipeline steps

1. Trims adapter sequences
2. Runs FastQC on the trimmed sequences for quality control of the samples, in parallel with the steps that follow
3. Maps reads to the selected reference genome
4. Adds UMI and gene information to the reads
5. Quantifies gene expression by counting reads
6. Counts UMI's for cases of PCR bias
7. Detects Differentially Expressed (DE) genes for a model with a single factor

Steps 3 and 5 are performed only for Mars-Seq

Step 6 is performed only if DESeq2 is selected



2.2 Pipeline report

Upon completion of the analysis, you will be sent an email with links to the results report.

The report includes several sections:

1. Sequencing and Mapping QC
 - a. **Figure 1** - Plots the average quality of each base across all reads. Qualities of 30 (predicted error rate 1:1000) and above are good
 - b. **Figure 2** - Histogram showing the number of reads for each sample in the raw data

- c. **Figure 3** - Histogram showing the percentage of reads discarded after trimming the adapters (after removing adapters, short, polyA/T and low quality reads are discarded by the pipeline)
 - d. **Figure 4** - Histogram with the number of reads for each sample in each step of the pipeline
 - e. **Figure 5** - Plots sequence coverage on and near gene regions (Not available in current version)
 - f. **Figure 6** -
 - i. Histogram showing the percentage of mapped reads (both uniquely and not uniquely) per sample
 - ii. Histogram showing the percentage of the uniquely mapped reads that mapped to genes (included genes must have at least 5 reads)
2. **Exploratory Analysis**
- a. **Figure 7** - Heatmap plotting the highly-expressed genes (above 5% of total expression). For example, the expression of gene RN45S in sample SRR3112243 constitutes 15% of the total expression
 - b. **Figure 8** - Heatmap of Pearson correlation between samples according to gene expression values
 - c. **Figure 9** - Clustering dendrogram of the samples according to gene expression
 - d. **Figure 10 - PCA analysis**
 - i. Histogram of % explained variability for each PC component
 - ii. PCA plot of PC1 vs PC2
 - iii. PCA plot of PC1 vs PC3
3. Differential Expression Analysis (this section exists only if you run the DESeq2 analysis) - a table with the number of differentially expressed genes (DE) in each category (up/down) for the different contrasts. In addition, links for p-value distribution, volcano plots and heatmaps, as well as a table of the DE genes with dot plots of their expression values are also provided
4. Bioinformatics Pipeline Methods - description of pipeline methods
5. Links to additional results - links for downloading tables with raw, normalized counts, log normalized values (rld), and statistical data of contrasts. In cases of models with batches, “combat” values were calculated (instead of rld) using the “sva” package, providing batch corrected normalized log2 count values.

2.3 Annotation file

For counts of the reads per gene, we use annotation files (gtf format) from “Ensembl” or “GENCODE”. In MARS-seq analysis, we extend the 3’ UTR exon away from the transcript on the DNA and extend or cut the 3” UTR exon towards the 5’ direction on the mRNA.

2.4 Examples of reports

[RNA-Seq example](#)

[Mars-seq example](#)

Note: This example analysis demonstrates a good starting point, and not necessarily an end result.

CHAPTER 3

Installation

Support: utap@weizmann.ac.il

A new UTAP version will be released soon on 2023

4.1 Registration to the system

UTAP - NGS PIPELINES User Datasets Run pipeline **Help**

Please login to see this page.


UTAP - ngs pipelines hosted on bbcu-weizmann-
demo server

Please Sign In

Username:

Password:

[forget password ?](#)

מכון ויצמן למדע  WEIZMANN INSTITUTE OF SCIENCE

Developed by Refael Kohen (refael.kohen@weizmann.ac.il)
Bioinformatics unit at Life Sciences Core Facilities (LSCF)
Weizmann Institute

Click on the signup button and fill out the form:

Sign Up

Username:
 Required. 150 characters or fewer. Letters, digits and @/./+/-/_ only.

Email:
 Required. Inform a valid email address.

First name:
 Required.

Last name:
 Required.

Password:

- Your password can't be too similar to your other personal information.
- Your password must contain at least 8 characters.
- Your password can't be a commonly used password.
- Your password can't be entirely numeric.

Password confirmation:
 Enter the same password as before, for verification.

4.2 User datasets

The “User datasets” screen contains the list of the user’s analyses. You can see the status of the run (RUNNING/SUCCESSFUL/FAILS). You need to refresh the page to see if the status has changed (you also will get email in the end of the run).

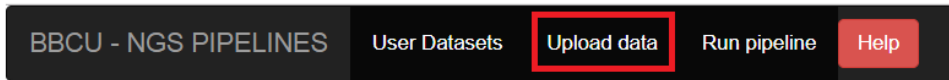
BBCU - NGS PIPELINES **User Datasets** Upload data Run pipeline Help

Analyses List:

	Name	Run status	Pipeline	Created	
<input type="button" value="Delete"/>	20180429_112922_MARS-seq-test	SUCCESSFUL	Transcriptome Mars-seq Deseq	April 29, 2018, 11:29 a.m.	
<input type="button" value="Delete"/>	20180214_131118_MARS-seq-test	SUCCESSFUL	Transcriptome Mars-seq	Feb. 14, 2018, 1:11 p.m.	<input type="button" value="Run Deseq again with other parameters"/>
<input type="button" value="Delete"/>	20180214_131005_RNA-seq-test	SUCCESSFUL	Transcriptome RNA-seq	Feb. 14, 2018, 1:10 p.m.	<input type="button" value="Run Deseq again with other parameters"/>

4.3 Import Input data

In order to run the transcriptome analysis pipeline, fastq sequence files need to be located on the server. Click on the “Upload data” button on the navigation bar, and select the folder of fastq files.



Upload data

Upload input folder to server.

Select directory: No file chosen

4.3.1 Instructions for uploading data to the server

Before running the pipeline, first upload the input file to the server as follows:

Press on the “choose files” button, and select the root folder of the files

Uploading folder of fastq files

- Fastq files must be organized, within the selected folder (root folder), into subfolders as shown below. Subfolders names are derived from sample names.
- Each subfolder contains the relevant fastq file, which can be compressed into the “gz” format.
- Fastq file names must end with “_R1.fastq(.gz)” or “_R1.fq(.gz)” for single-read data. The “R” prefix donates the read number.
- In the case of paired-end data (required for Mars-Seq), corresponding files must exist, with suffix “_R2.fastq(.gz)” instead of “_R1.fastq(.gz)”.

For example:

- **root folder**
 - **sample1**
 - * sample1_R1.fastq
 - * sample1_R2.fastq (must exist in Mars-seq and in paired-end)
 - **sample2**
 - * sample2_R1.fastq
 - * sample2_R2.fastq (must exist in Mars-seq and in paired-end)

The pipeline also supports the old convention of the fastq file names `_L00*_R1_0.fastq`. The letter “L” denotes the lane number, “R” denotes the read number, and the numbers 001,002 etc denote serial number of the file for each lane and read

For example:

- **root folder**
 - **sample1**
 - * `sample1_S0_L001_R1_001.fastq`
 - * `sample1_S0_L001_R1_002.fastq`
 - * `sample1_S0_L002_R1_001.fastq`
 - * `sample1_S0_L002_R1_002.fastq`
 - * `sample1_S0_L001_R2_001.fastq`
 - * `sample1_S0_L001_R2_002.fastq`
 - * `sample1_S0_L002_R2_001.fastq`
 - * `sample1_S0_L002_R2_002.fastq`
 - **sample2**
 - * ...

Uploading input for the demultiplexing pipeline

- For the pipeline of demultiplexing from BCL files: upload the original bcl folder. The original folder name should adhere to the template: `<date>_<field2>.<field3>_field4>`, e.g. `180514_NB551168_0123_AHTHHKBGX5`.
- For the pipeline of demultiplexing from FASTQ files: upload the fastq files. Note: the pipeline gets one file per read as input (i.e. one file for *R1*, *R2*, *II* etc.).

4.4 Run analysis

After importing you data (or if you have old data on the server that was imported in the past), you can run the pipeline by selecting the “Run pipeline” option

BBCU - NGS PIPELINES User Datasets Upload data **Run pipeline** Help

Run analysis

Choose pipeline from the list.

Choose pipeline:

----- ▼

- Transcriptome RNA-seq
- Transcriptome Mars-seq
- Demultiplexing_from_FASTQ
- Demultiplexing_from_BCL

4.4.1 Run RNA-seq or MARS-seq pipeline

RNA-seq Analysis Setup

If your protocol is RNA-seq, you will get this screen:

<p>Choose pipeline:</p> <p>----- ▼</p>	<p>Chosen pipeline:</p> <p>Project name:</p> <p>Input folder:</p> <p>Genome:</p> <p>Annotation:</p> <p>Output folder:</p> <p>User email:</p> <p>Stranded protocol:</p> <p>Adapter on R1 (default: True-Seq kit):</p> <p>Adapter on R2 (default: True-Seq kit):</p> <p>Deseq run:</p>	<p>Transcriptome RNA-seq</p> <p>Fill in project name</p> <p>Select input folder</p> <p>----- ▼</p> <p>----- ▼</p> <p>Select output folder</p> <p>Fill in your email</p> <p>stranded ▼</p> <p>AGATCGGAAGAGCACACGTCTGAACTCCAGTCAC</p> <p>AGATCGGAAGAGCGTCGTGTAGGGAAAGAGTGTAGATCTCGGTGGTCGCCGTATCATT</p> <p>No Deseq ▼</p>
	<p>Run analysis</p>	

If your protocol is MARS-seq, you will get this screen:

Choose pipeline:

Chosen pipeline: Transcriptome Mars-seq

Project name:

Input folder:

Genome:

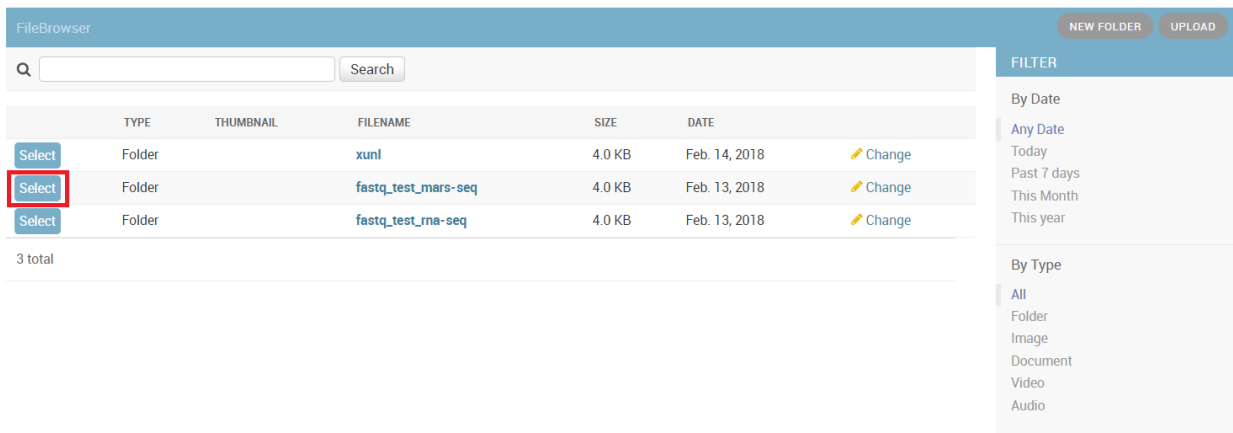
Annotation:

Output folder:

User email:

Deseq run:

In the input folder field, Browse within your directory structure and Select the **root folder** for analysis. Note that if you wish to go up one level (or more) click on the desired folder level using the path at the top of the window.



Input folder names must conform to the correct format as previously described. If there is a problem with the folder you selected, first resolve the error and then retry, selecting the updated folder.

If you wish the output folder to be different from the one automatically filled in (based on the selected input folder), just select the desired output folder.

Fill in the project name, then select the genome and annotation.

For RNA-seq protocols, choose whether your protocol is stranded (sequenced reads save the original strand of RNA fragments) or non-stranded.

Define the type of your adapters for each read (R1 and R2). These adapters will be removed from the reads by the pipeline. You can leave the default adapters if you use True-seq protocol P5 and P7 adapters.

To identify what's differentially expressed by using the DESeq2 package, select the Run Deseq option. By default, two categories must be created. Fill in the category names for each of the 2 categories shown. To define more categories, click on the Add Categories button to enable entering their details.

Deseq run: Run Deseq ▾

Add Category Remove Category Add Batch Effect

Filter samples (type part of the name)

- samp1test
- samp2test
- samp4test

Category 1 name

Category 2 name

Run analysis

Choose the samples by first selecting them, and then using the arrows to move them to the appropriate categories. You may also add additional categories.

Deseq run: Run Deseq ▾

Add Category Remove Category

Add Batch Effect

»»
>
<
«

»»
>
<
«

Filter samples (type part of the name)
 samp4test

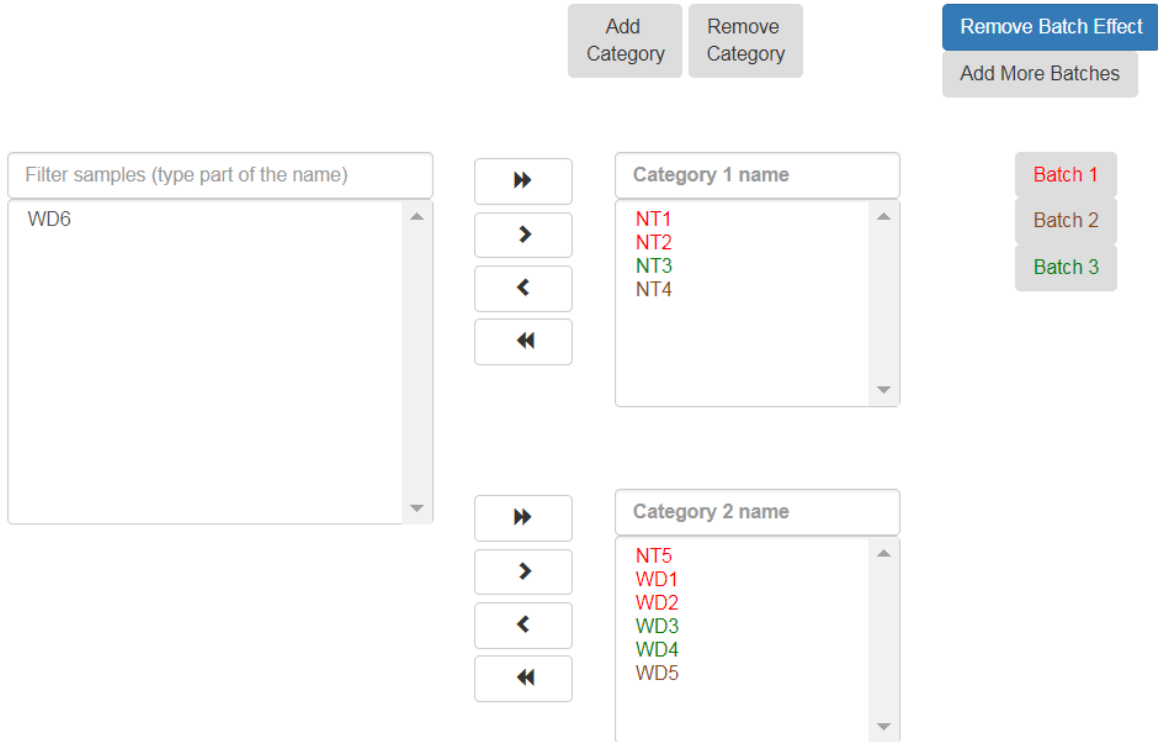
Treatment
 samp1test
 samp2test

Category 2 name

Run analysis

The order of what's being compared will be determined by the specification order of the categories. For example, DESeq2's output will show a "Treatment" vs "Control" comparison when "Treatment" is defined to be the first category, and "Control" the second.

If the samples were prepared in different batches, one can annotate them as follows: After moving the samples into category boxes, click on the "Add Batch Effect" button, select the samples from the category boxes that belong to a particular batch, and click on the "Batch 1" button. Repeat the operation for the other batches. Be sure that the batch effect is designed correctly - see DESeq2 documentation [here](#).



All of the steps of the pipeline (mapping, counts etc.) will be run on all of the samples, with the exception of Deseq which will be run only on samples with categories.

Finally, click on the “Run analysis” button.

At the end of the run, an email will be sent reporting analysis completion.

Using the pipeline efficiently

If you want re-run only the Deseq step several times on the same input folder (with other comparisons/batches), after completion of the initial analysis you will see (on the “user dataset” screen) a new button called “run again with other parameters”. Clicking on this button will re-run only the Deseq step.

Thus, the analysis will re-run only the short Dseq step (which takes a few minutes) and not re-run all of the time-consuming steps of the complete pipeline.

4.4.2 Run Demultiplexing pipeline

There are 2 pipelines for demultiplexing; the first accepts BCL files as input, the second fastq files.

Demultiplexing from BCL files

Upload BCL files to the server according to the following instructions:

All original BCL file folders must be built according to Next-seq (or Hi-seq) machine requirements. Folder names should adhere to the template <date>_<machine name>_<run number>_<flowcell id>, e.g. “170802_NB501465_0140_AH3W3KBGX3”.

The pipeline converts bcl files to fastq files, and demultiplexes fastq files according to MAR-seq or True-seq (or semi-True-seq) protocols.

Demultiplexing from fastq files

The pipeline demultiplexes fastq files according to MAR-seq or True-seq (or semi- True-seq) protocols.

Upload fastq files to the server according to the following instructions:

Note that the pipeline get as input one file per read (i.e. one file for each of *R1*, *R2*, *II* etc.). Choose the root folder of the fastq files from the list.

If the sequencing is single read, choose the file with *R1* in its file name for read 1, the file with either *R2* or *II* for index read, and leave the read 2 field empty.

If the sequencing is paired end, choose the file with *R1* in its file name for read 1, the file with *R2* for read 2, and the file with *II* for index read. If no file name includes *II*, choose one with *R2* for index read, and one with *R3* for read 2.

4.5 Customization

We chose the various pipeline parameters based on our rich experience in transcriptome analysis. This works very well for users who are not deeply familiar with bioinformatics software, and who prefer to quickly benefit from these choices without having to delve into the pipeline's architecture. On the other hand, many research groups have their own particular preferences, and can achieve flexibility by making some minor adjustments to the code as follows:

1. System-wide changes:

The Snakefiles of the pipelines are built using the Snakemake workflow management system (<https://snakemake.readthedocs.io>), and are located at:

```
$CONDA/envs/utap/lib/python2.7/site-packages/ngs-snakemake/snakefile-marseq.txt  
$CONDA/envs/utap/lib/python2.7/site-packages/ngs-snakemake/snakefile-rnaseq.txt
```

(where \$CONDA is the location of the miniconda - see <https://utap.readthedocs.io/en/latest/rst/installation.html>).

Users can modify the above scripts by adding new rules, commands and changing parameters.

In addition, one can customize the following R script's DESeq2 analyses and report generation:

```
$CONDA/envs/utap/lib/python2.7/site-packages/ngs-snakemake/reports.Rmd
```

After the code is changed, subsequent analyses using the web application will reflect these changes.

2. Ad-hoc changes:

Another option is to change parameters only for a particular run.

After running the analysis in the usual way, one can navigate to the output folder, which contains a copy of the snakefile and Rmd script. One can then change and re-run the analysis from the linux terminal by executing:

```
./snakemake_cmd_RUN_ID (where an example of a RUN_ID is a timestamp like "20171205_145424").
```


CHAPTER 5

Releases

Docker version:

- 1.0.7 - Added installation parameter to determine maximum cores that UTAP can use.
- 1.0.6 – Added installation parameter to support a variety Portable Batch System (PBS) clusters
- 1.0.5 – Support for the ATAC-seq pipeline

Pipeline version:

You can find the version of the run under the output folder of each analysis

- 1.0.67 - FDR correction function is disabled

6.1 UTAP source code

<https://bitbucket.org/bbcu/utap>

6.2 Dependencies

Softwares

Genomes

CHAPTER 7

License

UTAP is licensed under GNU General Public License version 3. License needed for commercial use.

CHAPTER 8

Author

Refael Kohen (until version 1.0.7),
refael.kohen@weizmann.ac.il, refael.kohen@gmail.com
support: utap@weizmann.ac.il
Bioinformatics unit at Life Sciences Core Facilities (LSCF)
Weizmann Institute of Science, Rehovot 76100, Israel.

CHAPTER 9

Acknowledgments

Please cite: Kohen R, Barlev J, Hornung G, Stelzer G, Feldmesser E, Kogan K, Safran M, Leshkowitz D: UTAP: User-friendly Transcriptome Analysis Pipeline. BMC Bioinformatics 2019, 20(1):154.