STAGE 0: data description

Link to instance: https://ribogalaxy.genomicsdatascience.ie/ . First, if you want to keep your progress and have a bigger data storage quota, you need to create an account on RiboGalaxy.

Input file is named: '*SAMPLE.fq*'. It's ribosome profiling of human U-2 OS cells and it's a subsample of ~30k reads that were selected to map on 12 transcripts. It was demultiplexed (file contains reads from 1 sample only), however barcodes are still there.

It can be downloaded : https://www.dropbox.com/s/h8xrtx7taxblzll/SAMPLE.fq?dl=0. Note, that gzipped fastq files are also accepted.

In this tutorial we assume that you had some previous basic experience working with any Galaxy instance, e.g. you know how to run tools and how to look at resulting files. If you don't, the interface is pretty intuitive and can be learnt on the fly.

This tutorial also contains steps that link processed ribo-seq data with GWIPS-viz (https://gwips.ucc.ie/) and Trips-viz (http://trips.ucc.ie/) browsers for further analysis and visualisation, however it does not contain the full tutorial for those tools. If you're interested in exploring your data in detail using their full functionality you can take a look at corresponding publications and a walkthrough example for Trips-viz https://trips.ucc.ie/help/?parent_acc=parent_walkthrough. If you are familiar with the UCSC Genome browser, GWIPS-viz would be very easy to use.

Now let's move to the read composition.

Here is the structure of a read:
UUU - **RPF sequence** - **NNNNN** - **BBBBB** – AGATCGGAAGAGCACACGTCTGAA

UUU = untemplated additions
RPF sequence = ribosome protected fragment sequence
NNNNN = UMIs
BBBBB = barcodes
AGATCGGAAGAGCACACGTCTGAA = 3' adapter

Depending on a protocol, the structure of a read may vary and you need to adjust your parameters accordingly. This tutorial deals with exactly this structure.

Raw reads in fastq file looks like that:

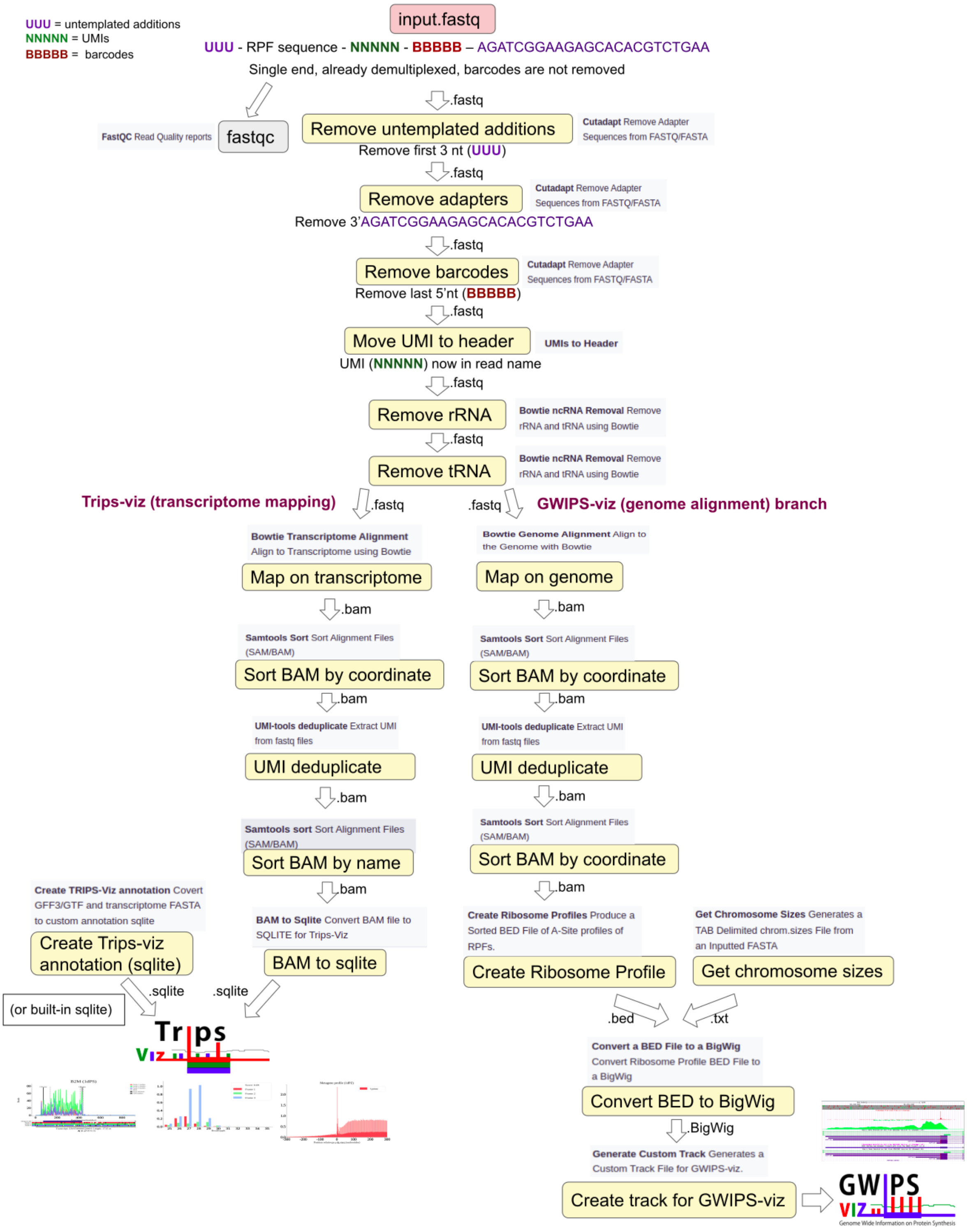
```
@A01174:295:H7CWGDSX3:4:2548:4824:16736 1:N:0:ATTCAGAA+AGGCTATA
CCCCCGGGGCTACGCCTGTCTGAGCGTCGCTTGATGCTAGAAGATCGGAAGAGCACACGTCT
GAACTCCAGTCACATTCAGAAATCTCGTATGCCGTCTTCTGCTTGAAAAAGGGGGGGGGGGGG
GGGGGGGGGGGGGGGGGGGGGGGGG
```

```
+
FFFFFFFFFFFFFF:FFFFFFFFFFFFFFFFF,FFFF,F,FF:FF:FFFF:FFFFFFFFF:F,FF:,FFF:FFFFF,F,F:
FFF:F,,F:F,,FF:::FF,F:F,F:FFFFF,FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF
@A01174:295:H7CWGDSX3:4:1652:1533:7091 1:N:0:ATTCAGAA+AGGCTATA
CCGCGTGGGGGGCCCAAGTCCTTCTGATCGAGGCCCTTGGCTAGAAGATCGGAAGAGCACAC
GTCTGAACTCCAGTCACATTCAGAAATCTCGTATGCCGTCTTCTGCTTGAAAAGGGGGGGGGG
GGGGGGGGGGGGGGGGGGGGGGGG
+
FFFFFFFFFFFFFFFFF:FFFF:FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF
FFFFFFFFFFFFFFFFFFFFFFF,FF:FFFF:::F:FFFFF:,FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF
F
```

Most tools have a wide range of parameters that can be set.

Text descriptions and screenshots will show you a minimal set of parameters you need to define. Parameters that are not mentioned in text or on screenshots can be skipped. However, If you know what you’re doing or want to experiment, feel free to adjust parameters to your liking.

Tutorial looks very long and intimidating, however, the sample is small and most steps can be done pretty quickly (except creating sqlite annotation and uploading it to Trips-viz in STAGE IIa, though it is provided in dropbox).



UUU = untemplated additions
NNNNN = UMIs
BBBBB = barcodes

input.fastq

UUU - RPF sequence - NNNNN - BBBBB – AGATCGGAAGAGCACACGTCTGAA

Single end, already demultiplexed, barcodes are not removed

FastQC Read Quality reports — fastqc

.fastq

Remove untemplated additions — Cutadapt Remove Adapter Sequences from FASTQ/FASTA

Remove first 3 nt (UUU)

.fastq

Remove adapters — Cutadapt Remove Adapter Sequences from FASTQ/FASTA

Remove 3’AGATCGGAAGAGCACACGTCTGAA

.fastq

Remove barcodes — Cutadapt Remove Adapter Sequences from FASTQ/FASTA

Remove last 5’nt (BBBBB)

.fastq

Move UMI to header — UMIs to Header

UMI (NNNNN) now in read name

.fastq

Remove rRNA — Bowtie ncRNA Removal Remove rRNA and tRNA using Bowtie

.fastq

Remove tRNA — Bowtie ncRNA Removal Remove rRNA and tRNA using Bowtie

Trips-viz (transcriptome mapping)

.fastq

Bowtie Transcriptome Alignment Align to Transcriptome using Bowtie

Map on transcriptome

.bam

Samtools Sort Sort Alignment Files (SAM/BAM)

Sort BAM by coordinate

.bam

UMI-tools deduplicate Extract UMI from fastq files

UMI deduplicate

.bam

Samtools sort Sort Alignment Files (SAM/BAM)

Sort BAM by name

.bam

BAM to Sqlite Convert BAM file to SQLITE for Trips-Viz

BAM to sqlite

.sqlite

Create TRIPS-Viz annotation Covert GFF3/GTF and transcriptome FASTA to custom annotation sqlite

Create Trips-viz annotation (sqlite)

.sqlite

(or built-in sqlite)

Trips Viz

GWIPS-viz (genome alignment) branch

.fastq

Bowtie Genome Alignment Align to the Genome with Bowtie

Map on genome

.bam

Samtools Sort Sort Alignment Files (SAM/BAM)

Sort BAM by coordinate

.bam

UMI-tools deduplicate Extract UMI from fastq files

UMI deduplicate

.bam

Samtools Sort Sort Alignment Files (SAM/BAM)

Sort BAM by coordinate

.bam

Create Ribosome Profiles Produce a Sorted BED File of A-Site profiles of RPFs.

Create Ribosome Profile

.bed

Get Chromosome Sizes Generates a TAB Delimited chrom.sizes File from an Inputted FASTA

Get chromosome sizes

.txt

Convert a BED File to a BigWig Convert Ribosome Profile BED File to a BigWig

Convert BED to BigWig

.BigWig

Generate Custom Track Generates a Custom Track File for GWIPS-viz.

Create track for GWIPS-viz

GWIPS VIZ
Genome Wide Information on Protein Synthesis

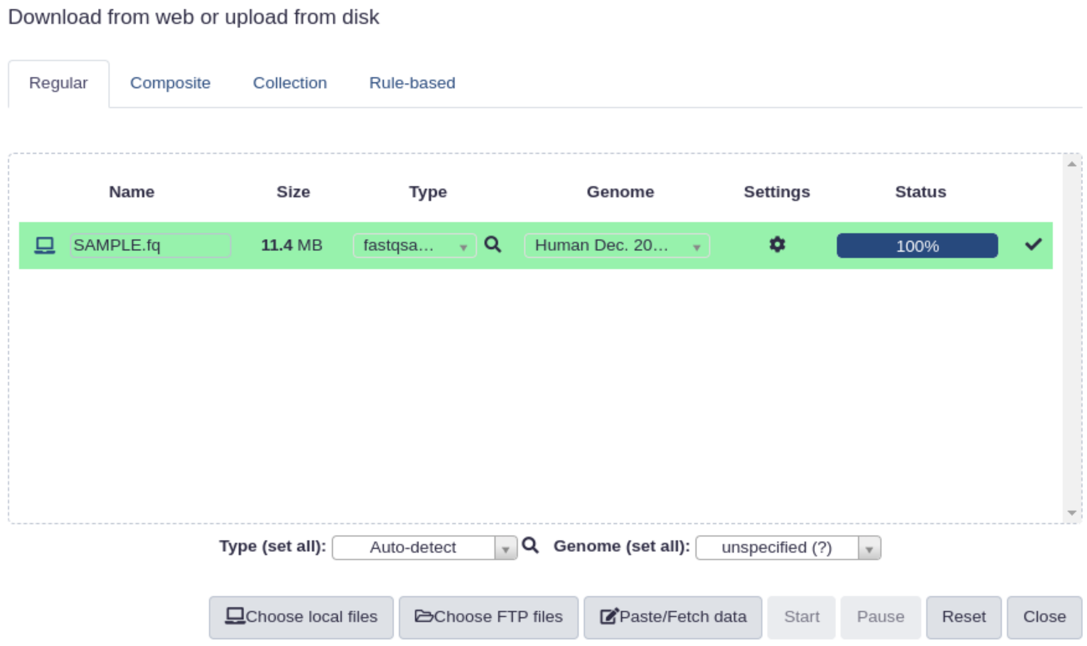
STAGE I: preprocessing and basic QC. First we need to retrieve the Ribosome Protected Fragment sequence from the raw read.

Just to give you a quick overview about the key steps and tools that we will need to use for obtaining clean Ribosome Protected Fragments from raw reads:

- After raw data is uploaded, we would want to look at the quality of the reads using the pretty standard tool called ***FastQC***.
- After that we will trim adapters, untemplated additions and barcodes using ***Cutadapt***.
- Next, we will move UMIs from the read sequence to the name (header) of the read using ***UMIs to Header*** tool.
- Then we will deal with typical major contaminants of ribosome profiling - tRNA and rRNAs by using ***Bowtie ncRNA Removal***.
- At the end we can again check the quality of the resulting data using ***FastQC***.

Let's begin!

1. Upload sample to the RiboGalaxy ('**Upload data**' at the left panel), choose Type 'fastqsanger' and Genome - 'hg38'. Push 'Start'.



Download from web or upload from disk

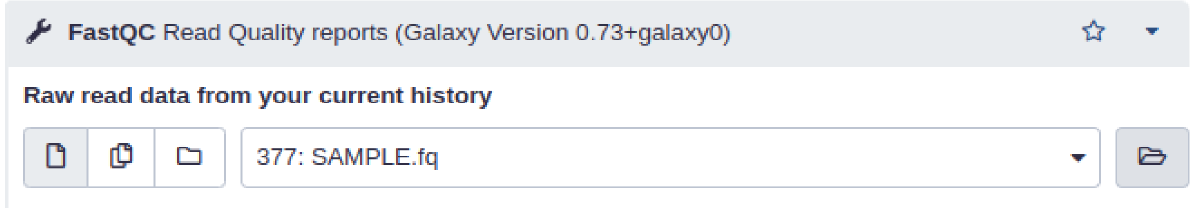
Regular | Composite | Collection | Rule-based

Name	Size	Type	Genome	Settings	Status
SAMPLE.fq	**11.4** MB	fastqsa...	Human Dec. 20...	⚙	100% ✓

Type (set all): Auto-detect **Genome (set all):** unspecified (?)

Choose local files | Choose FTP files | Paste/Fetch data | Start | Pause | Reset | Close

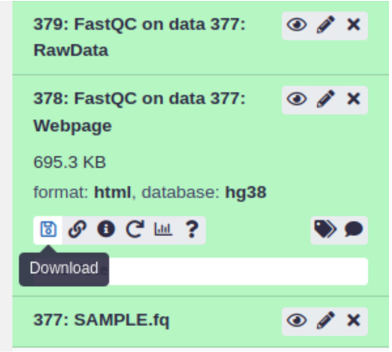
2. Run quality control using the **Fastqc** tool from the **Preprocessing** section. Push 'Execute'.



FastQC Read Quality reports (Galaxy Version 0.73+galaxy0)

Raw read data from your current history

377: SAMPLE.fq



379: FastQC on data 377: RawData

378: FastQC on data 377: Webpage

695.3 KB

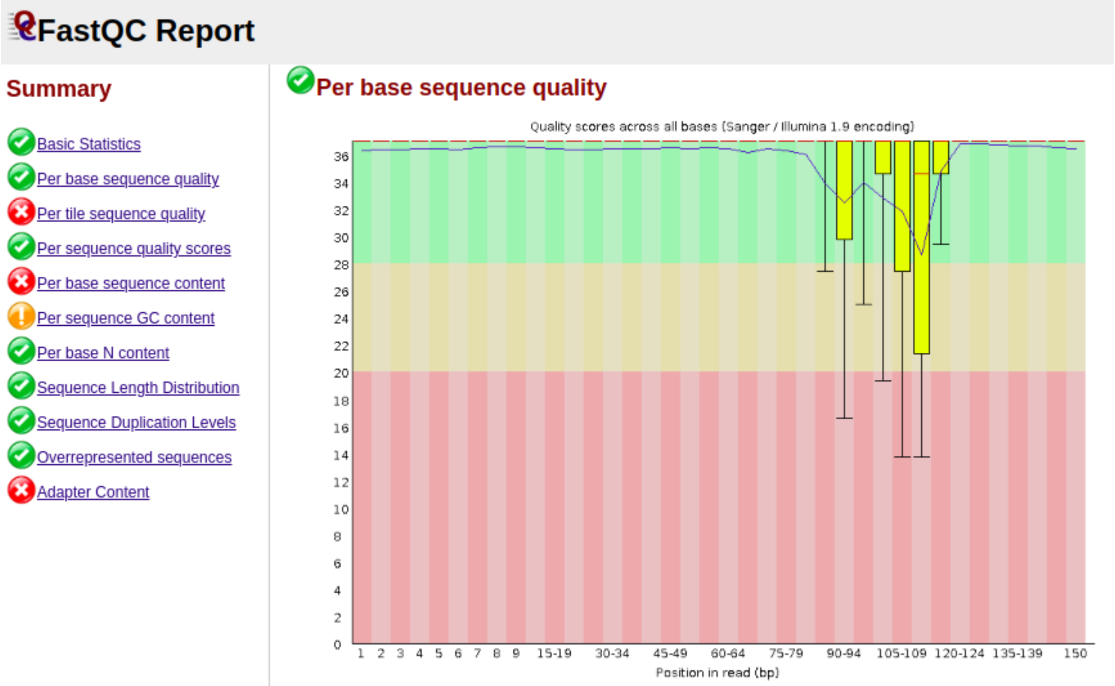
format: **html**, database: **hg38**

377: SAMPLE.fq

It outputs Webpage and RawData on your right panel. You can download the Webpage by clicking on the ‘save’ icon. It will download the zip archive containing the file SAMPLE_fq_fastqc.html which you can open in a browser by clicking on it.

You can explore different quality control plots, especially ones that are failed or have warnings. Let’s have a look at the passed QC plot: **Per base sequence quality**. X-axis shows position in read (bp), y-axis shows distribution of sequencing quality score per base for all reads which is split by 3 areas: green is high quality, yellow is acceptable, red is poor quality. Here the entire read has either high or acceptable quality. You may sometimes observe that the end or the beginning of the read has poor quality and therefore it requires trimming. Or even the entire read sequence has poor quality - therefore you may consider filtering reads based on quality or even choose another dataset to work with…

Here we have not yet removed adapters, barcodes, untemplated additions and UMIs, so after all the steps read will be much shorter (~28-30nt).



FastQC Report

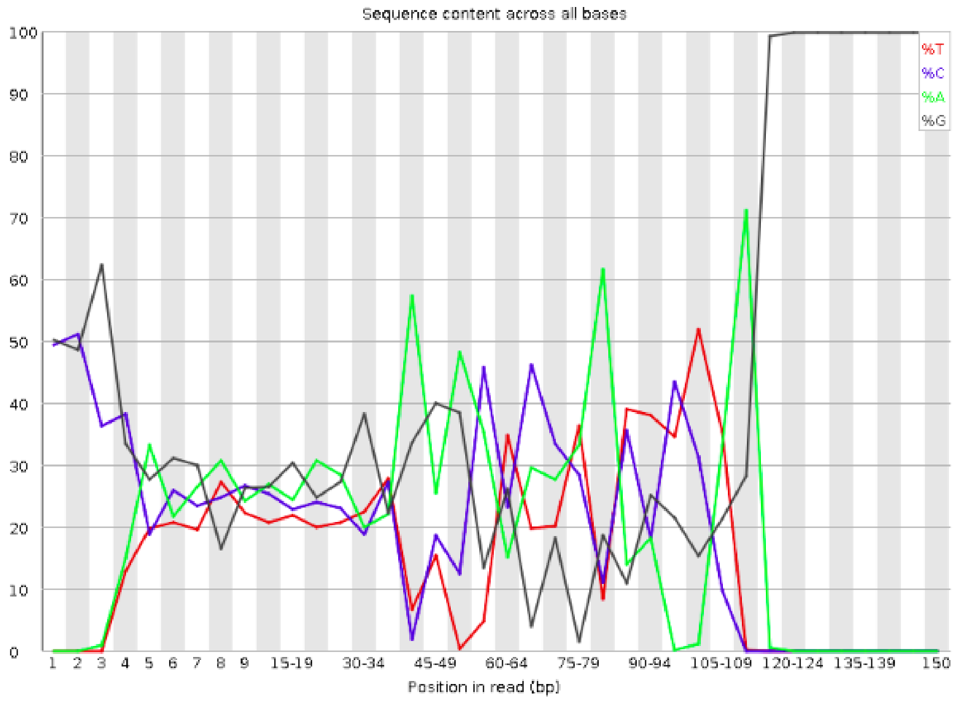
Summary

- ✓ Basic Statistics
- ✓ Per base sequence quality
- ✗ Per tile sequence quality
- ✓ Per sequence quality scores
- ✗ Per base sequence content
- ! Per sequence GC content
- ✓ Per base N content
- ✓ Sequence Length Distribution
- ✓ Sequence Duplication Levels
- ✓ Overrepresented sequences
- ✗ Adapter Content

✓ Per base sequence quality

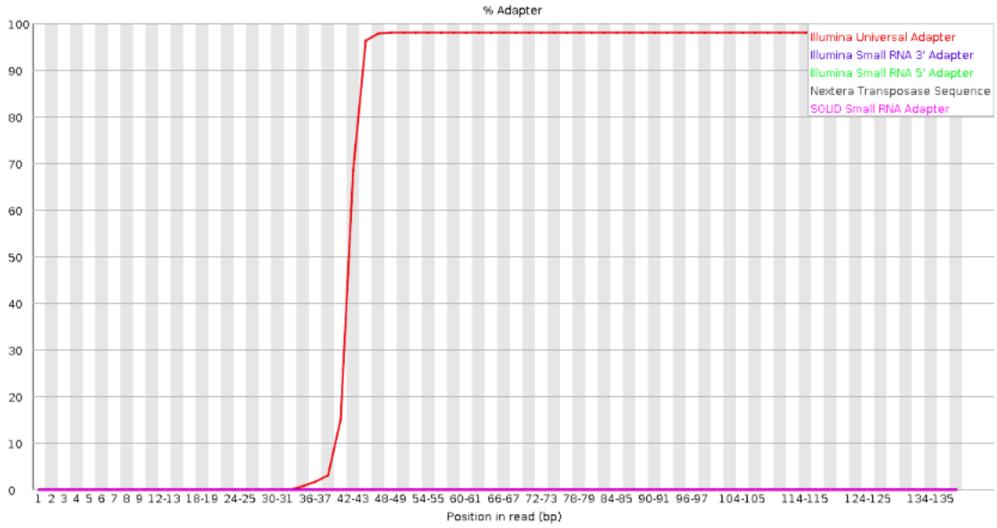
Let’s look at the failed one: **Per base sequence content**. Y-axis is the fraction of nucleotide per position in read, x-axis is position in read (bp). We have not yet removed untemplated additions (first 3 nucleotides), adapter (last dozens of nucleotides), 5nt of barcode preceding the adapter and 5nt of UMI preceding the barcode. Indeed, RPF starting from position 4 till approximately position 30-34 has relatively uniform distribution of nucleotides as expected, while adapters, barcodes, UMIs and untemplated additions show clearly huge spread in fractions of nucleotides. We will look at this plot once we remove all of that.

Per base sequence content



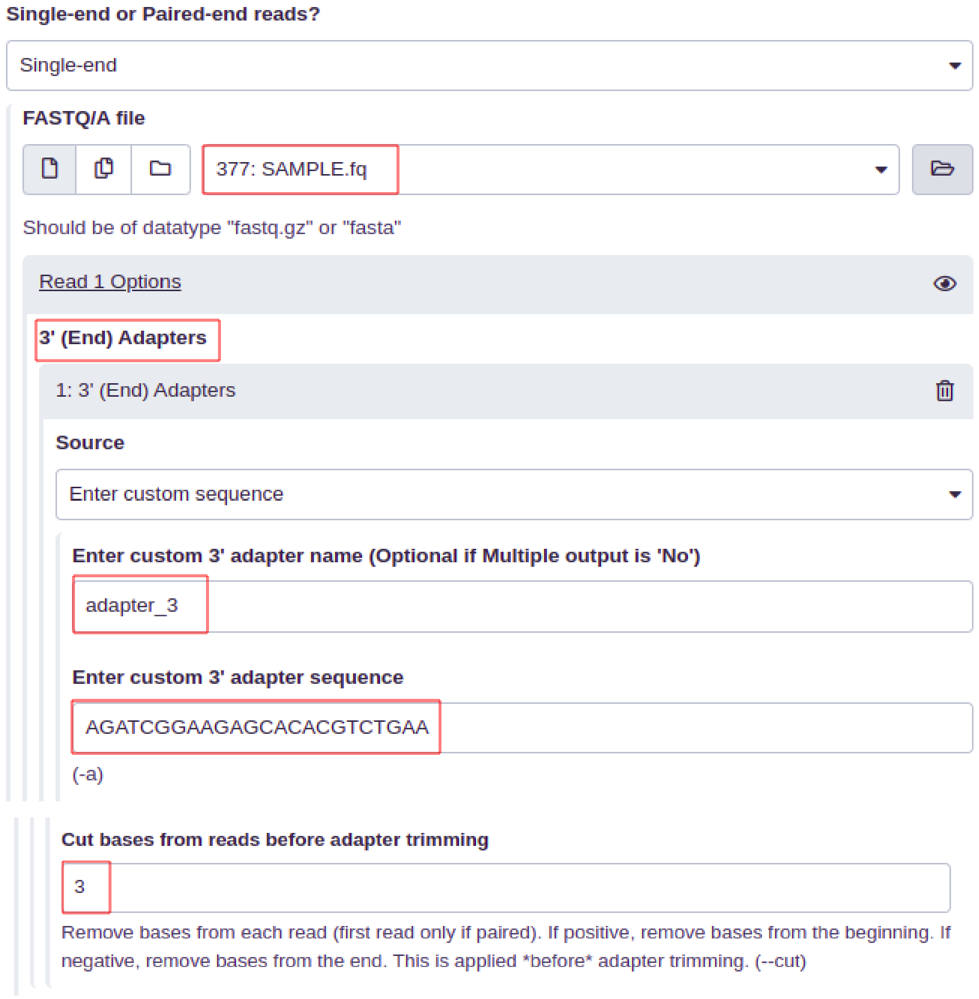
We do also see untrimmed adapters in **Adapter Content** plot (also **failed** one, y-axis is the cumulative fraction of of reads where the adapter sequence is identified at the indicated base position and x-axis is position in read in bp):

Adapter Content



3. Trim untemplated additions (first 3nt - **UUU**) and adapter with ***Cutadapt*** from **Processing**. Choose the input file that you uploaded. We will trim the 3**’ (End) Adapter**. **Source**: Enter Custom Sequence. **Enter a custom 3’ adapter sequence**: AGATCGGAAGAGCACACGTCTGAA. If you’d like, you can also add the name of the adapter (in the screenshot below it’s ‘*adapter_3*’), but it’s not necessary since we

have only 1 adapter. **Cut bases from reads before adapter trimming**: choose 3 (those untemplated additions in the beginning of the read).



Single-end or Paired-end reads?

Single-end

FASTQ/A file

377: SAMPLE.fq

Should be of datatype "fastq.gz" or "fasta"

Read 1 Options

3' (End) Adapters

1: 3' (End) Adapters

Source

Enter custom sequence

Enter custom 3' adapter name (Optional if Multiple output is 'No')

adapter_3

Enter custom 3' adapter sequence

AGATCGGAAGAGCACACGTCTGAA

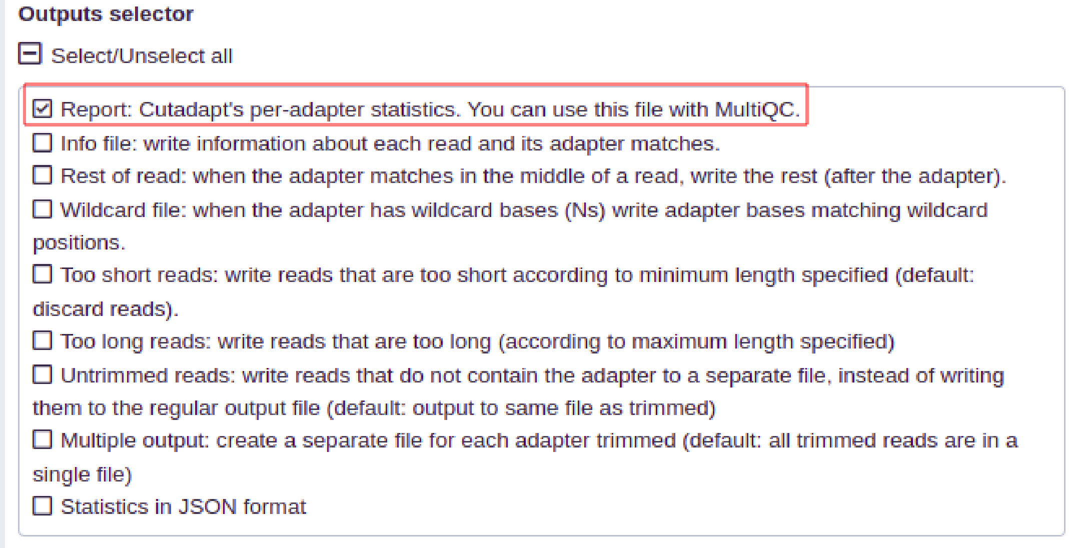
(-a)

Cut bases from reads before adapter trimming

3

Remove bases from each read (first read only if paired). If positive, remove bases from the beginning. If negative, remove bases from the end. This is applied *before* adapter trimming. (--cut)

Also you can choose additional files in the **Outputs selector**, e.g. here we will have ***Report*** for adapter statistics (e.g. how many reads contain adapters). Push 'Execute'. After this step read looks like: **RPF sequence** - **NNNNN** - **BBBBB**.

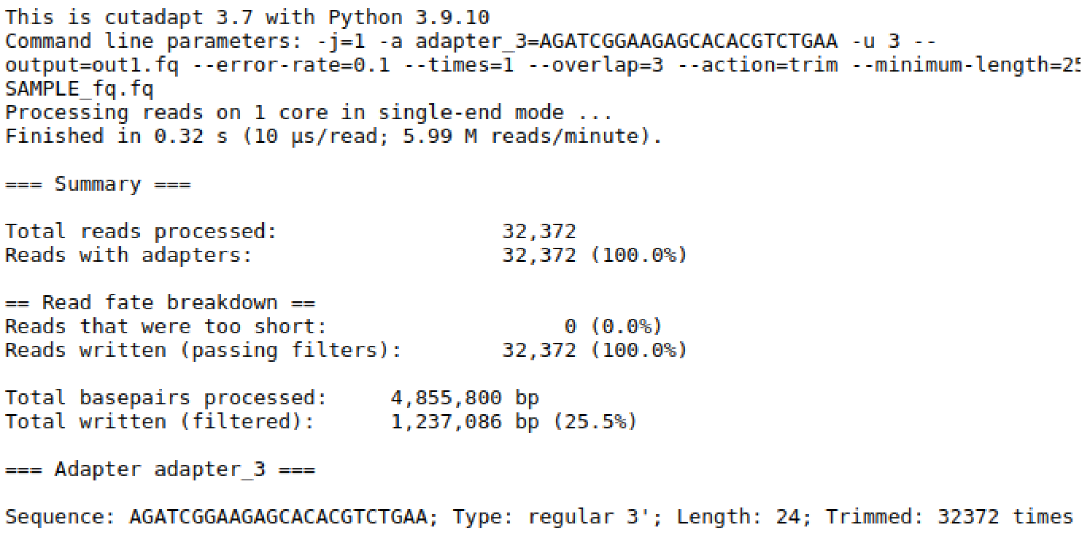


Outputs selector

⊟ Select/Unselect all

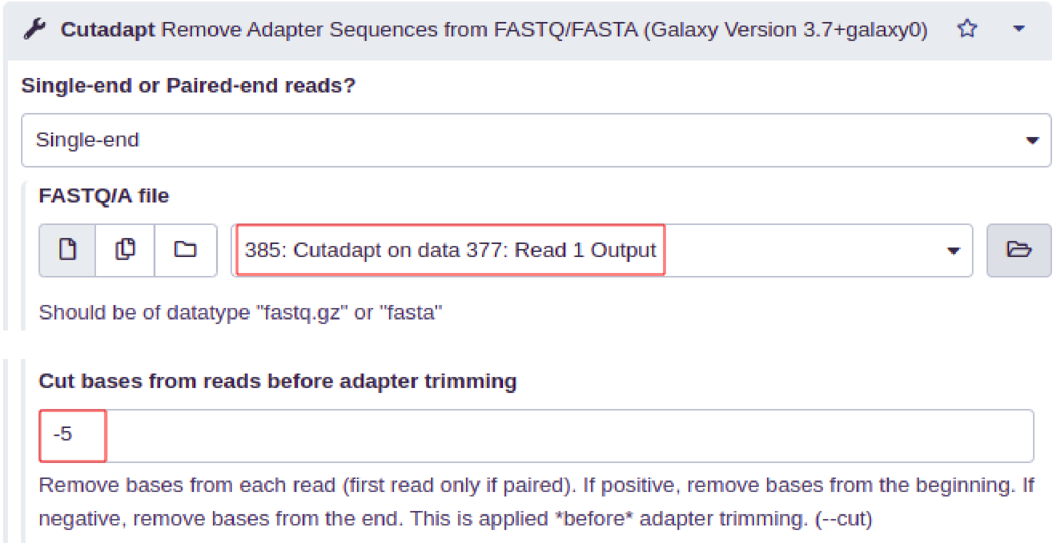
☑ Report: Cutadapt's per-adapter statistics. You can use this file with MultiQC.
☐ Info file: write information about each read and its adapter matches.
☐ Rest of read: when the adapter matches in the middle of a read, write the rest (after the adapter).
☐ Wildcard file: when the adapter has wildcard bases (Ns) write adapter bases matching wildcard positions.
☐ Too short reads: write reads that are too short according to minimum length specified (default: discard reads).
☐ Too long reads: write reads that are too long (according to maximum length specified)
☐ Untrimmed reads: write reads that do not contain the adapter to a separate file, instead of writing them to the regular output file (default: output to same file as trimmed)
☐ Multiple output: create a separate file for each adapter trimmed (default: all trimmed reads are in a single file)
☐ Statistics in JSON format

We can take a look at the report file (hit an 'eye' icon). All reads contain adapters as expected.



```
This is cutadapt 3.7 with Python 3.9.10
Command line parameters: -j=1 -a adapter_3=AGATCGGAAGAGCACACGTCTGAA -u 3 --
output=out1.fq --error-rate=0.1 --times=1 --overlap=3 --action=trim --minimum-length=2
SAMPLE_fq.fq
Processing reads on 1 core in single-end mode ...
Finished in 0.32 s (10 µs/read; 5.99 M reads/minute).

=== Summary ===

Total reads processed:                  32,372
Reads with adapters:                    32,372 (100.0%)

== Read fate breakdown ==
Reads that were too short:                   0 (0.0%)
Reads written (passing filters):        32,372 (100.0%)

Total basepairs processed:     4,855,800 bp
Total written (filtered):      1,237,086 bp (25.5%)

=== Adapter adapter_3 ===

Sequence: AGATCGGAAGAGCACACGTCTGAA; Type: regular 3'; Length: 24; Trimmed: 32372 times
```

4. Remove barcodes with the ***Cutadapt*** tool from the **Preprocessing section**. Input - fastq file from the previous step. Here you need to define the only one parameter: **Cut bases from reads before adapter trimming**. Choose -5 (trim 5 nt from the end of the read). Push ‘Execute’. Read will look like: **RPF sequence - NNNNN**.



Cutadapt Remove Adapter Sequences from FASTQ/FASTA (Galaxy Version 3.7+galaxy0)

Single-end or Paired-end reads?

Single-end

FASTQ/A file

385: Cutadapt on data 377: Read 1 Output

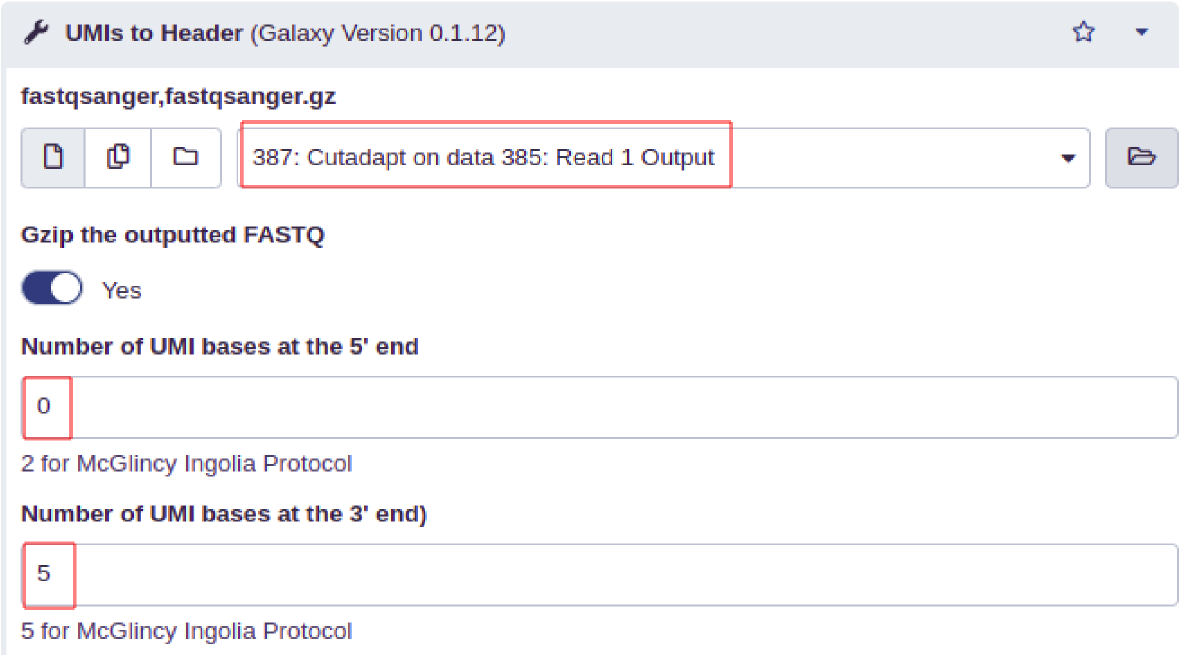
Should be of datatype "fastq.gz" or "fasta"

Cut bases from reads before adapter trimming

-5

Remove bases from each read (first read only if paired). If positive, remove bases from the beginning. If negative, remove bases from the end. This is applied *before* adapter trimming. (--cut)

5. Move UMIs to the header of the read. Use toll ***UMI to Header*** from **UMI and barcodes**. Input - .fastq file from the previous step. Choose ‘**Number of UMI bases at the 5’end**’ to 0 and ‘**Number of UMI bases at the 3’end**’ to 5. Push ‘Execute’. After this step read will look like: **RPF sequence**.



UMIs to Header (Galaxy Version 0.1.12)

fastqsanger,fastqsanger.gz

387: Cutadapt on data 385: Read 1 Output

Gzip the outputted FASTQ

Yes

Number of UMI bases at the 5' end

0

2 for McGlincy Ingolia Protocol

Number of UMI bases at the 3' end)

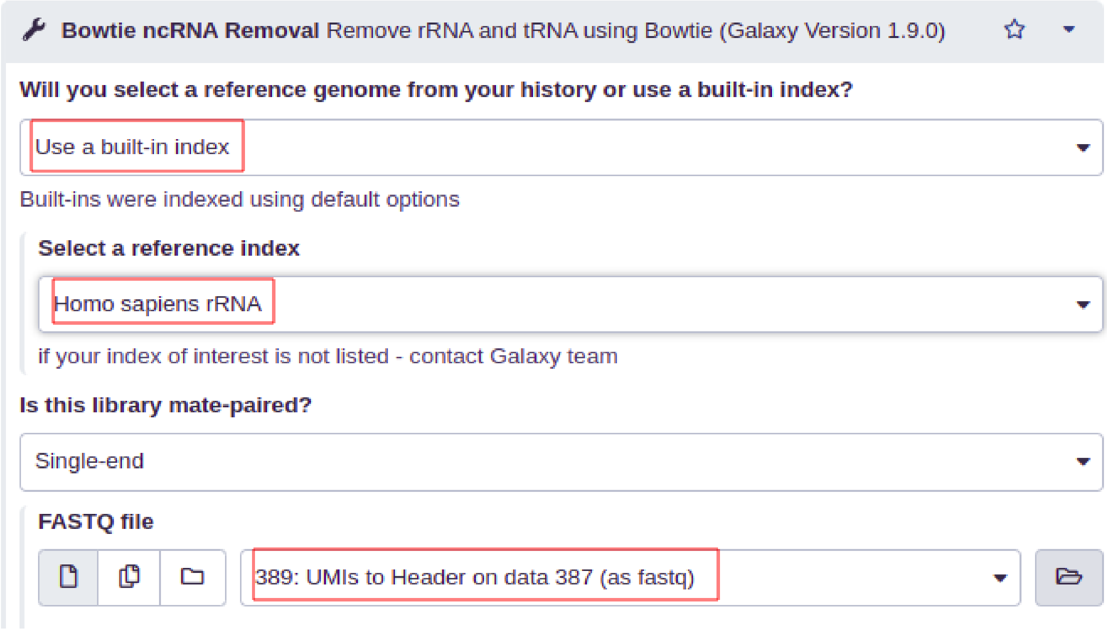
5

5 for McGlincy Ingolia Protocol

Notice that 5nt UMIs are now in the read header (‘_GTGGA’)

```
@A01174:295:H7CWGDSX3:4:1101:25446:1908_GTGGA 1:N:0:ATTCAGAA+AGGCTATA
GAACCACCCTGACAGCAGGGCTATAAT
+
FFFFFFFFFFFFFFFFFFFFFFFFFFF
```

6. Remove rRNA from the reads using the Bowtie ***ncRNA Removal tool*** from the **Preprocessing** section. Since its human reads, we can choose built-in index *Homo sapiens rRNA*. Input file is fastq file from the previous step. Push ‘Execute’.



Bowtie ncRNA Removal Remove rRNA and tRNA using Bowtie (Galaxy Version 1.9.0)

Will you select a reference genome from your history or use a built-in index?

Use a built-in index

Built-ins were indexed using default options

Select a reference index

Homo sapiens rRNA

if your index of interest is not listed - contact Galaxy team

Is this library mate-paired?

Single-end

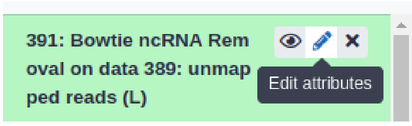
FASTQ file

389: UMIs to Header on data 387 (as fastq)

Let’s have a look at the report file (named ‘mapping stats’, hit an ‘eye’ icon): most reads do not align to rRNA (99.96%). We know that this sample is very clean thus we don’t expect to see a lot of rRNA reads. However, usually we expect to see ~80%

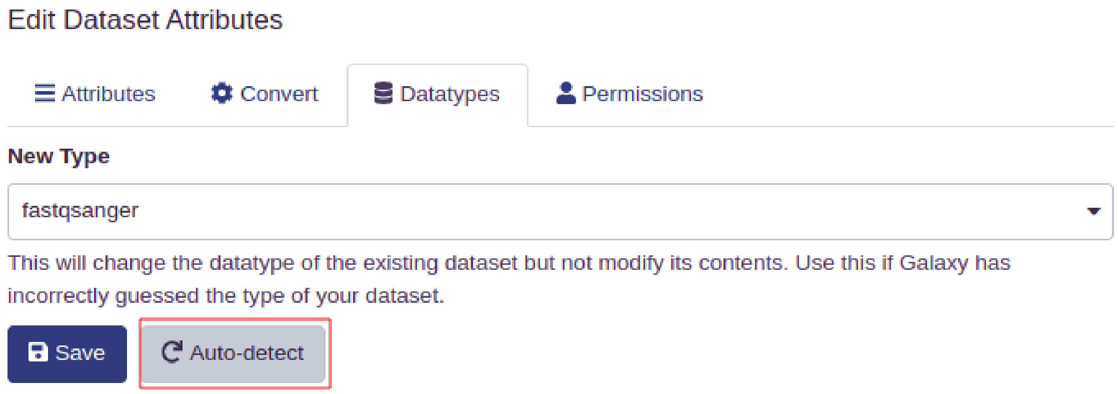
or even more of reads that map to rRNA in typical ribo-seq samples.

```
# reads processed: 32372
# reads with at least one reported alignment: 13 (0.04%)
# reads that failed to align: 32359 (99.96%)
Reported 13 alignments to 1 output stream(s)
```



391: Bowtie ncRNA Removal on data 389: unmapped reads (L)

Sometimes galaxy bowtie does not correctly output the datatype of unmapped reads that we are going to use for the next step. We can do the following trick. Choose ‘***Edit attributes***’ on the right panel and in the middle panel choose the ‘***Datatypes***’ tab and then ‘***Auto-detect***’. It should change the datatype to the right one, ‘*fastqsanger*’ (or you can type it manually).

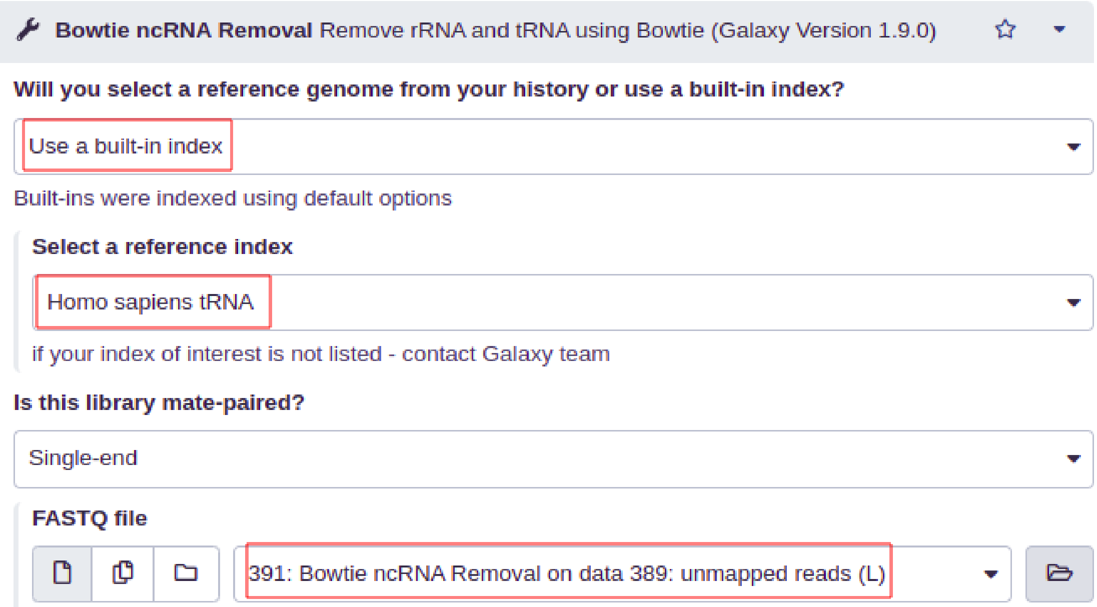


Edit Dataset Attributes

New Type: fastqsanger

This will change the datatype of the existing dataset but not modify its contents. Use this if Galaxy has incorrectly guessed the type of your dataset.

7. Remove tRNA from the reads using ***Bowtie ncRNA Removal tool*** from **Preprocessing**. Since its human reads, we can choose built-in index *Homo sapiens tRNA*. Input file is fastq file from the previous step with unmapped reads. Push ‘Execute’.



Bowtie ncRNA Removal Remove rRNA and tRNA using Bowtie (Galaxy Version 1.9.0)

Will you select a reference genome from your history or use a built-in index?: Use a built-in index

Built-ins were indexed using default options

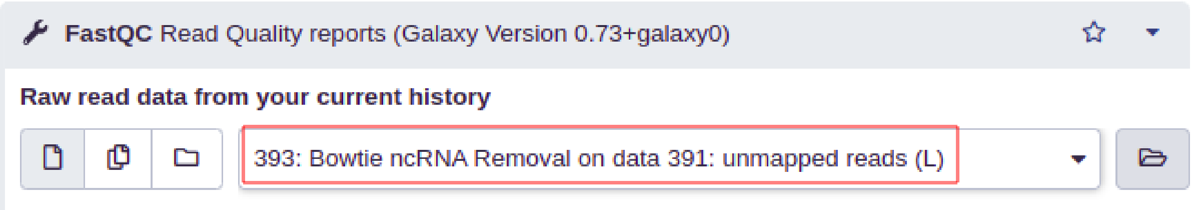
Select a reference index: Homo sapiens tRNA

if your index of interest is not listed - contact Galaxy team

Is this library mate-paired?: Single-end

FASTQ file: 391: Bowtie ncRNA Removal on data 389: unmapped reads (L)

8. Finally, we can repeat QC using the **FastQC** tool in **Preprocessing**. Input file is trimmed reads (no additions, adapters, UMIs and barcodes) filtered out from rRNA/tRNA contamination taken from step 7. Push ‘Execute’.



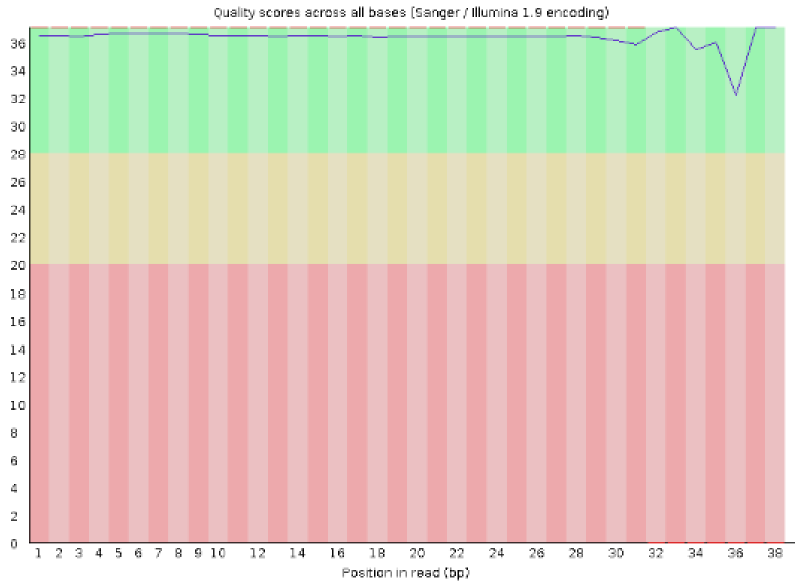
FastQC Read Quality reports (Galaxy Version 0.73+galaxy0)

Raw read data from your current history

393: Bowtie ncRNA Removal on data 391: unmapped reads (L)

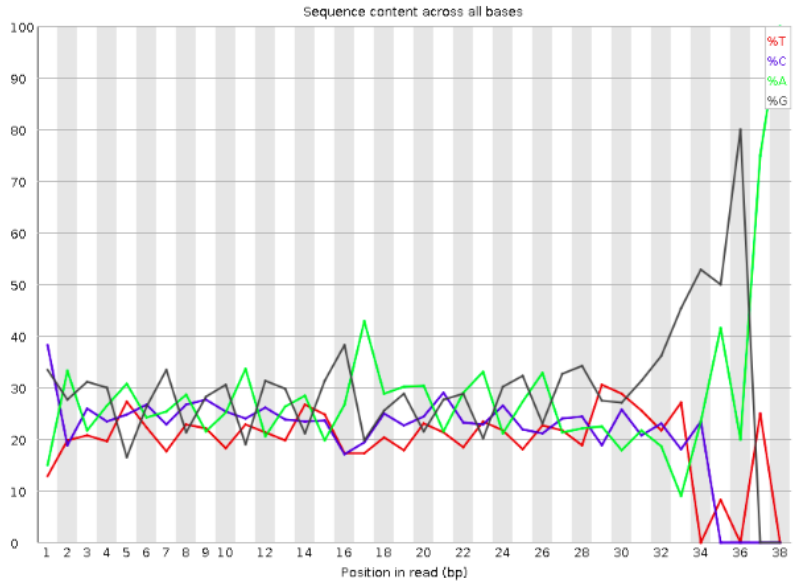
Let’s examine the report .html file. **Per base sequence quality** is perfect:

✓ **Per base sequence quality**



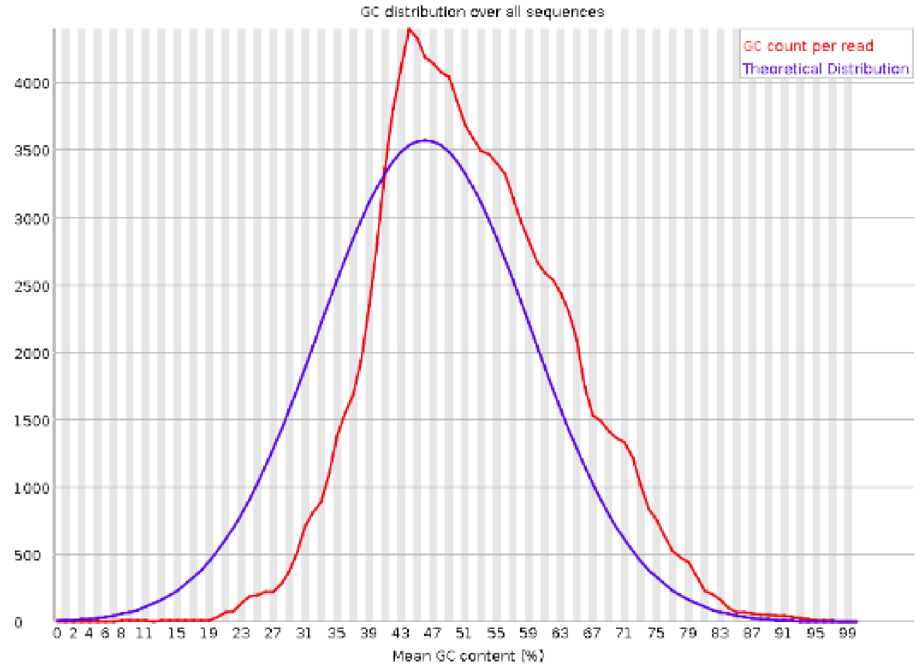
Per base sequence content looks much better then for the raw reads, however 3’end of reads is still wobbly. We are not going to trim it since for deriving A-sites of ribosomes we use 5’end of reads.

✗ **Per base sequence content**



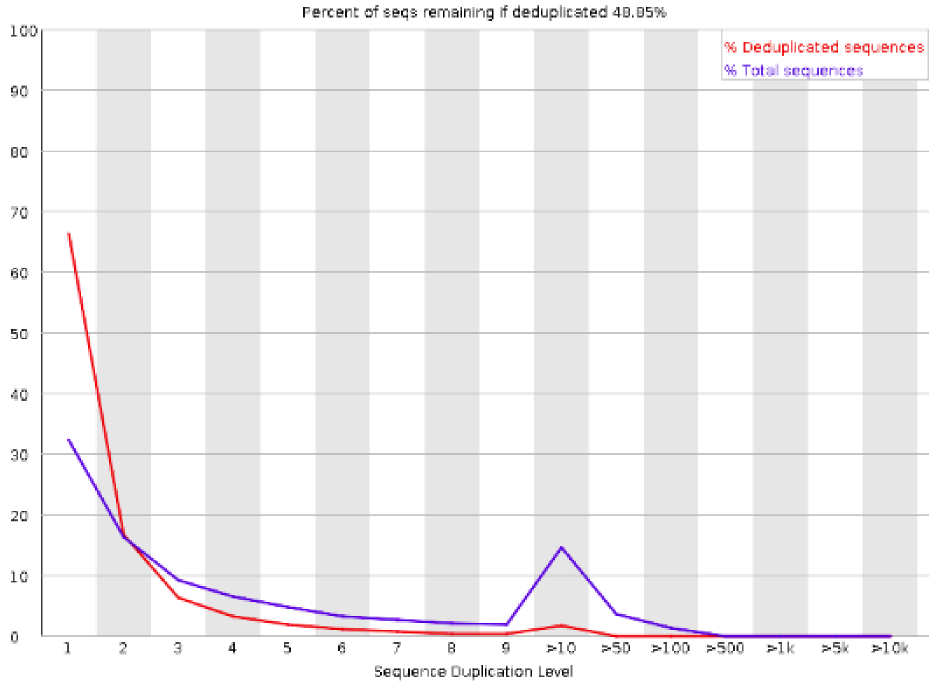
Per sequence GC content. It shows GC content across the whole length of each sequence in a file and compares it to a modelled normal distribution of GC content. In our case, GC content is significantly different from expected one which can be explained by non-random sampling of reads (reads that map to only 12 transcripts).

Per sequence GC content

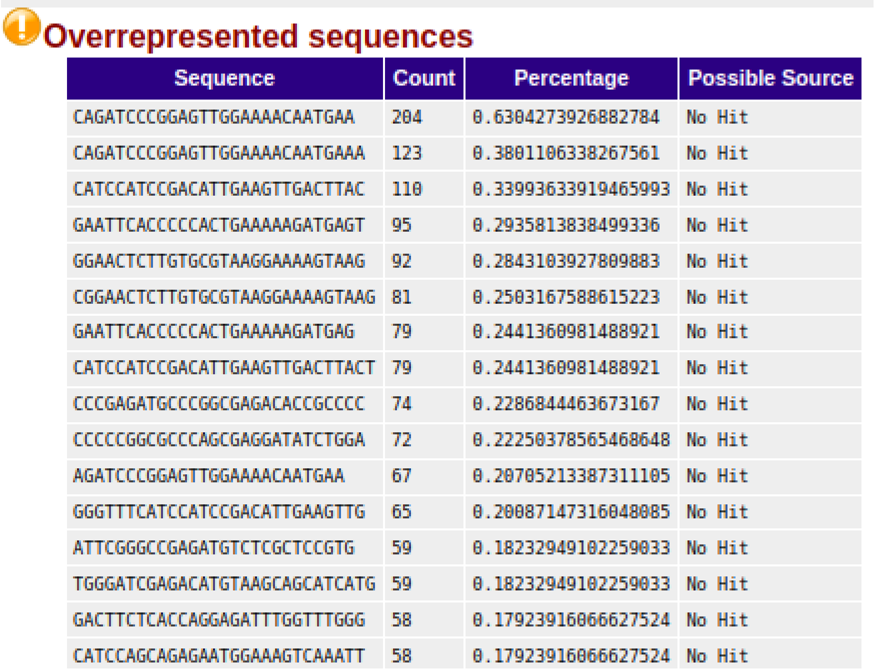


Sequence Duplication Levels. X-axis is how often a sequence occurs (1 = one time, 2 = twice, >10 = more than 10 times etc), y-axis is the fraction of such sequences (reads). Typically in RNA-seq and Ribo-seq experiments it’s expected to see many duplicated reads.

Sequence Duplication Levels



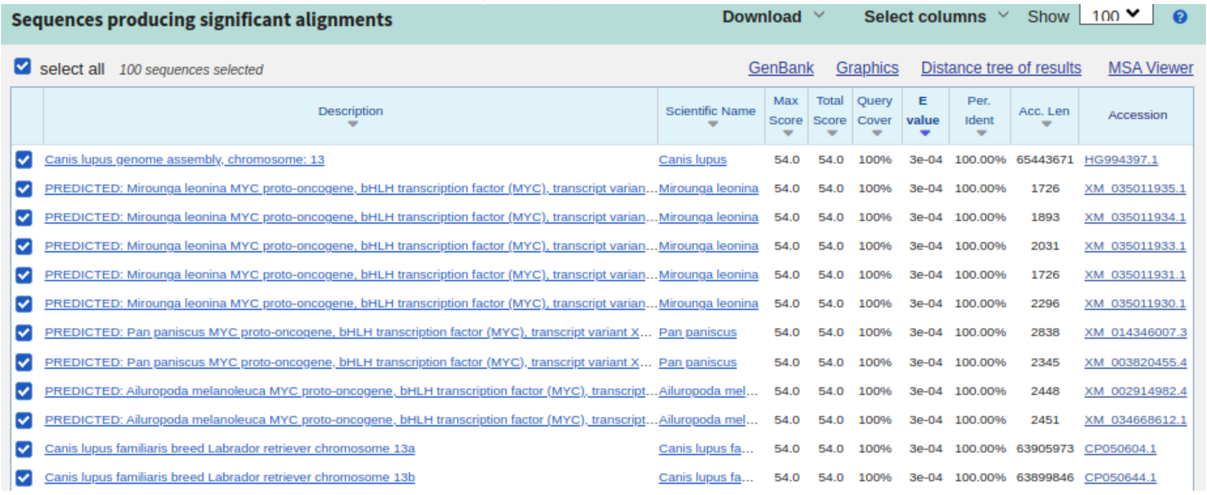
It’s also good to take a look at **Overrepresented sequences** and blast top hits. This way we can spot unexpected contaminations. We had about 32k reads and the top overrepresented sequence is less than 1%.



Overrepresented sequences

Sequence	Count	Percentage	Possible Source
CAGATCCCGGAGTTGGAAAACAATGAA	204	0.6304273926882784	No Hit
CAGATCCCGGAGTTGGAAAACAATGAAA	123	0.3801106338267561	No Hit
CATCCATCCGACATTGAAGTTGACTTAC	110	0.33993633919465993	No Hit
GAATTCACCCCCACTGAAAAAGATGAGT	95	0.2935813838499336	No Hit
GGAACTCTTGTGCGTAAGGAAAAGTAAG	92	0.2843103927809883	No Hit
CGGAACTCTTGTGCGTAAGGAAAAGTAAG	81	0.2503167588615223	No Hit
GAATTCACCCCCACTGAAAAAGATGAG	79	0.2441360981488921	No Hit
CATCCATCCGACATTGAAGTTGACTTACT	79	0.2441360981488921	No Hit
CCCGAGATGCCCGGCGAGACACCGCCCC	74	0.2286844463673167	No Hit
CCCCCGGCGCCCAGCGAGGATATCTGGA	72	0.22250378565468648	No Hit
AGATCCCGGAGTTGGAAAACAATGAA	67	0.20705213387311105	No Hit
GGGTTTCATCCATCCGACATTGAAGTTG	65	0.20087147316048085	No Hit
ATTCGGGCCGAGATGTCTCGCTCCGTG	59	0.18232949102259033	No Hit
TGGGATCGAGACATGTAAGCAGCATCATG	59	0.18232949102259033	No Hit
GACTTCTCACCAGGAGATTTGGTTTGGG	58	0.17923916066627524	No Hit
CATCCAGCAGAGAATGGAAAGTCAAATT	58	0.17923916066627524	No Hit

When we blast the top1 sequence, we can see that it’s part of MYC which is indeed among transcripts we sampled originally.



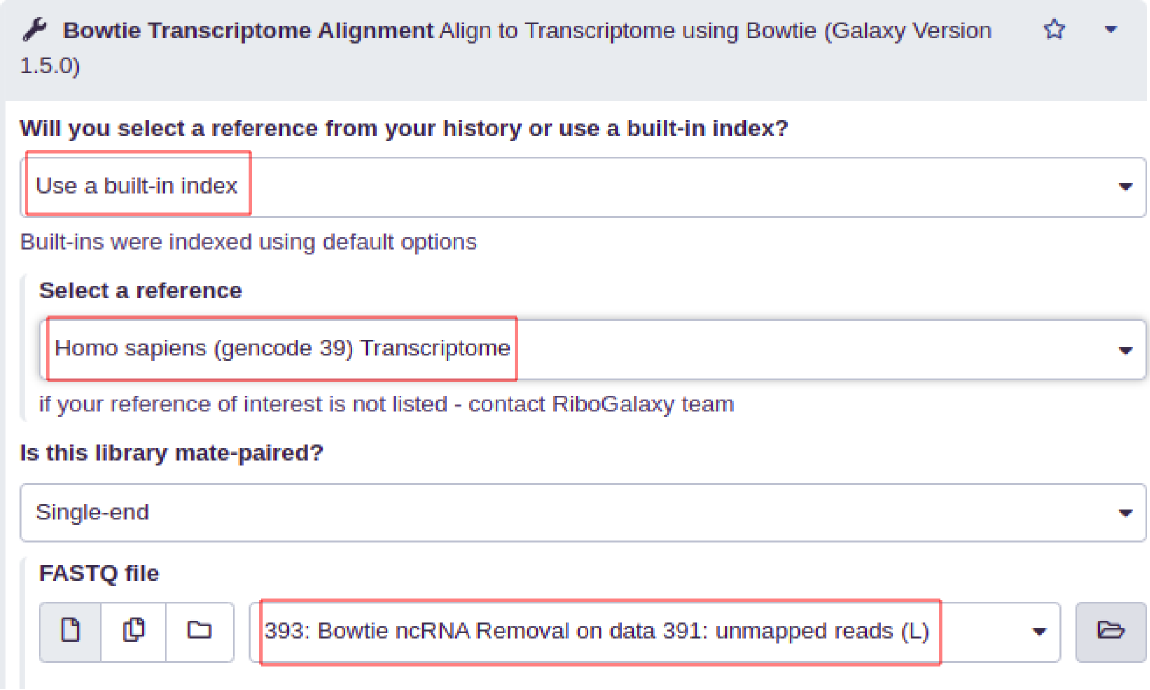
	Description	Scientific Name	Max Score	Total Score	Query Cover	E value	Per. Ident	Acc. Len	Accession
☑	Canis lupus genome assembly, chromosome: 13	Canis lupus	54.0	54.0	100%	3e-04	100.00%	65443671	HG994397.1
☑	PREDICTED: Mirounga leonina MYC proto-oncogene, bHLH transcription factor (MYC), transcript varian...	Mirounga leonina	54.0	54.0	100%	3e-04	100.00%	1726	XM_035011935.1
☑	PREDICTED: Mirounga leonina MYC proto-oncogene, bHLH transcription factor (MYC), transcript varian...	Mirounga leonina	54.0	54.0	100%	3e-04	100.00%	1893	XM_035011934.1
☑	PREDICTED: Mirounga leonina MYC proto-oncogene, bHLH transcription factor (MYC), transcript varian...	Mirounga leonina	54.0	54.0	100%	3e-04	100.00%	2031	XM_035011933.1
☑	PREDICTED: Mirounga leonina MYC proto-oncogene, bHLH transcription factor (MYC), transcript varian...	Mirounga leonina	54.0	54.0	100%	3e-04	100.00%	1726	XM_035011931.1
☑	PREDICTED: Mirounga leonina MYC proto-oncogene, bHLH transcription factor (MYC), transcript varian...	Mirounga leonina	54.0	54.0	100%	3e-04	100.00%	2296	XM_035011930.1
☑	PREDICTED: Pan paniscus MYC proto-oncogene, bHLH transcription factor (MYC), transcript variant X...	Pan paniscus	54.0	54.0	100%	3e-04	100.00%	2838	XM_014346007.3
☑	PREDICTED: Pan paniscus MYC proto-oncogene, bHLH transcription factor (MYC), transcript variant X...	Pan paniscus	54.0	54.0	100%	3e-04	100.00%	2345	XM_003820455.4
☑	PREDICTED: Ailuropoda melanoleuca MYC proto-oncogene, bHLH transcription factor (MYC), transcript...	Ailuropoda mel...	54.0	54.0	100%	3e-04	100.00%	2448	XM_002914982.4
☑	PREDICTED: Ailuropoda melanoleuca MYC proto-oncogene, bHLH transcription factor (MYC), transcript...	Ailuropoda mel...	54.0	54.0	100%	3e-04	100.00%	2451	XM_034668612.1
☑	Canis lupus familiaris breed Labrador retriever chromosome 13a	Canis lupus fa...	54.0	54.0	100%	3e-04	100.00%	63905973	CP050604.1
☑	Canis lupus familiaris breed Labrador retriever chromosome 13b	Canis lupus fa...	54.0	54.0	100%	3e-04	100.00%	63899846	CP050644.1

STAGE IIa: mapping on a transcriptome.

Let's talk briefly about what to expect at this stage.

- First, we will map clean reads (ribosome protected fragments) from preprocessing STAGE I on the transcriptome using ***Bowtie Transcriptome Alignment.***
- Next, we will sort the resulting alignments by coordinate because it is needed for the next stage where we get rid of PCR duplicates using UMIs. We will use ***Samtools sort***.
- Then we will deduplicate our sample based on UMIs using ***UMI-tools deduplicate***.
- Next, we will sort reads by name in the deduplicated sample using S***amtools sort*** since this order is required for building sqlite files. Sqlite is a format that is used to store mapped reads and associated statistics (e.g. triplet periodicity and ambiguity of mapping) for downstream analysis and visualisation in Trips-viz browser.
- Finally, we will create a sqlite file using the ***BAM to Sqlite*** tool. We will then upload it to Trips-viz and explore subcodon profiles.

1. We will map clean reads (RPFs) from Stage I on transcriptome using **Bowtie Transcriptome Alignment** from **Trips-viz (transcriptome mapping) branch**. In this example we will use the built-in index *Homo sapiens (gencode 39) Transcriptome* and the output fastq file from the STAGE I. Push 'Execute'.



Bowtie Transcriptome Alignment Align to Transcriptome using Bowtie (Galaxy Version 1.5.0)

Will you select a reference from your history or use a built-in index?

Use a built-in index

Built-ins were indexed using default options

Select a reference

Homo sapiens (gencode 39) Transcriptome

if your reference of interest is not listed - contact RiboGalaxy team

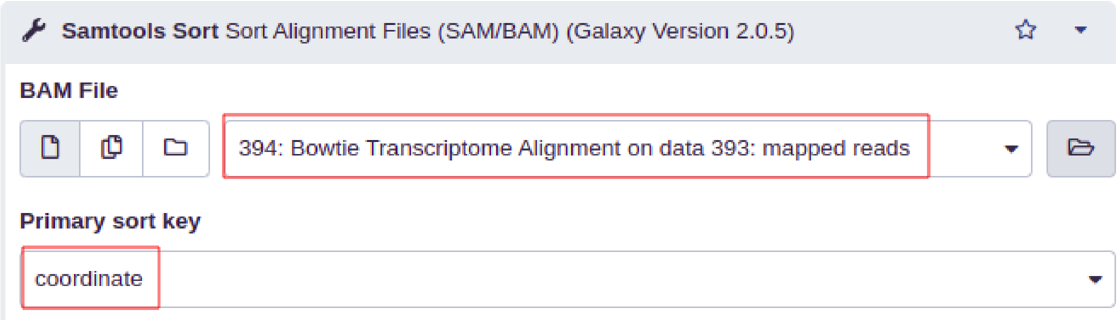
Is this library mate-paired?

Single-end

FASTQ file

393: Bowtie ncRNA Removal on data 391: unmapped reads (L)

2. Sort the bam file by coordinate using **Samtools sort** from **GWIPS-Viz (genomic alignment) branch**. Push 'Execute'.



Samtools Sort Sort Alignment Files (SAM/BAM) (Galaxy Version 2.0.5)

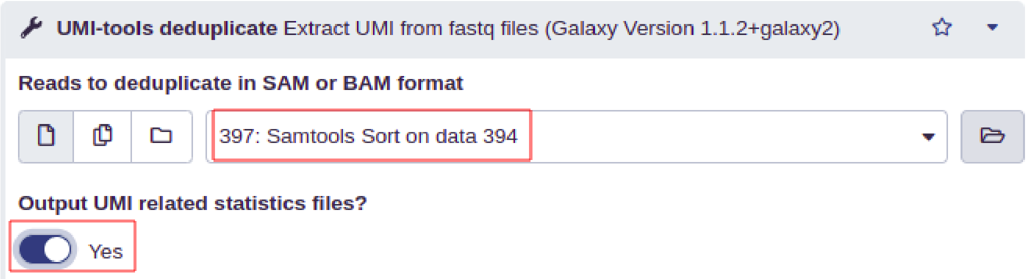
BAM File

394: Bowtie Transcriptome Alignment on data 393: mapped reads

Primary sort key

coordinate

3. This step can be performed if you want to deduplicate your sample (remove PCR duplicates) using UMIs. It can be done with **UMI-tools deduplicate** from **UMI and barcodes**. Input file is bam sorted by coordinate.You can also choose to output UMI related statistics file. Push 'Execute'.



UMI-tools deduplicate Extract UMI from fastq files (Galaxy Version 1.1.2+galaxy2)

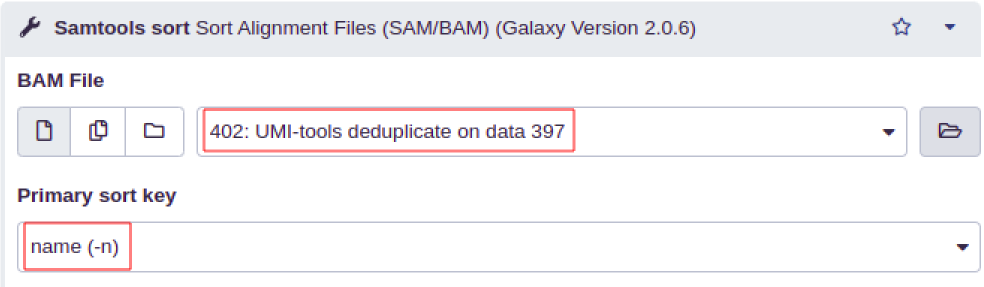
Reads to deduplicate in SAM or BAM format

397: Samtools Sort on data 394

Output UMI related statistics files?

Yes

4. Sort bam file by name using **Samtools sort** from **Trips-viz (transcriptome mapping) branch.** Push 'Execute'.



Samtools sort Sort Alignment Files (SAM/BAM) (Galaxy Version 2.0.6)

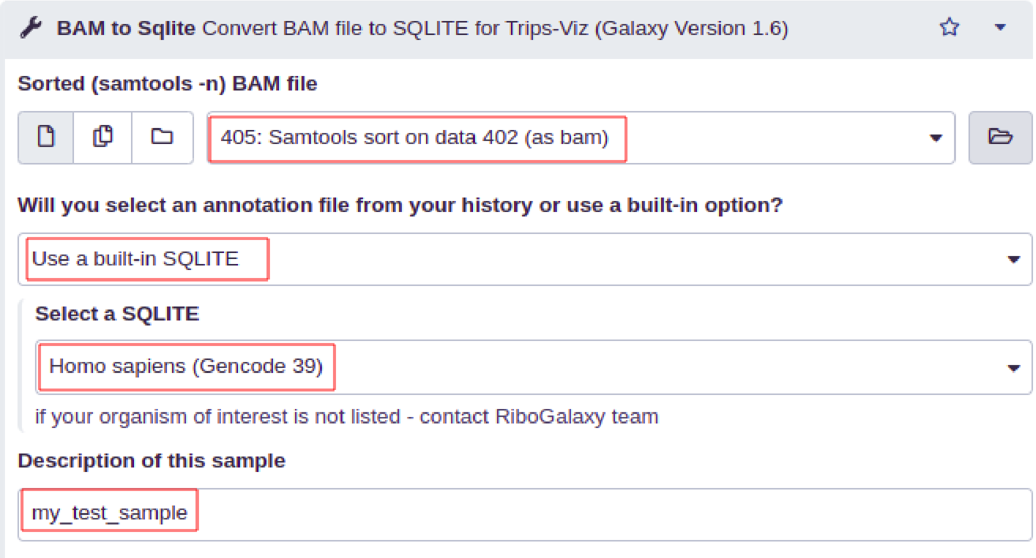
BAM File

402: UMI-tools deduplicate on data 397

Primary sort key

name (-n)

5. FInally we will get a sqlite file using sorted by name bam file and annotation sqlite file (we will choose built-in option: Gencode 39). We will use the **BAM to Sqlite** tool from the **Trips-viz (transcriptome mapping) branch**. We also can add a description of the sample, e.g. '*my_test_sample*'.



BAM to Sqlite Convert BAM file to SQLITE for Trips-Viz (Galaxy Version 1.6)

Sorted (samtools -n) BAM file

405: Samtools sort on data 402 (as bam)

Will you select an annotation file from your history or use a built-in option?

Use a built-in SQLITE

Select a SQLITE

Homo sapiens (Gencode 39)

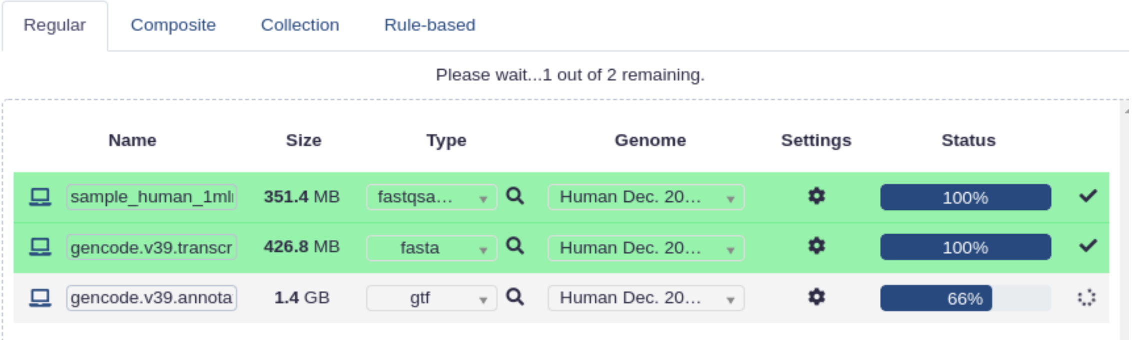
if your organism of interest is not listed - contact RiboGalaxy team

Description of this sample

my_test_sample

5' (***Optional, you can skip this step since an annotation sqlite file is provided as a built-in option and can be downloaded from dropbox***). Otherwise you can build annotation sqlite yourself (note that it will take a while) using **Create TRIPS-Viz annotation** from **Trips-viz (transcriptome mapping) branch**. For that you first need to download files and *unzip them*, and then upload to the RiboGalaxy:

- https://ftp.ebi.ac.uk/pub/databases/gencode/Gencode_human/release_39/gencode.v39.annotation.gtf.gz - .gtf file
- https://ftp.ebi.ac.uk/pub/databases/gencode/Gencode_human/release_39/gencode.v39.transcripts.fa.gz - .fasta file



Please wait...1 out of 2 remaining.

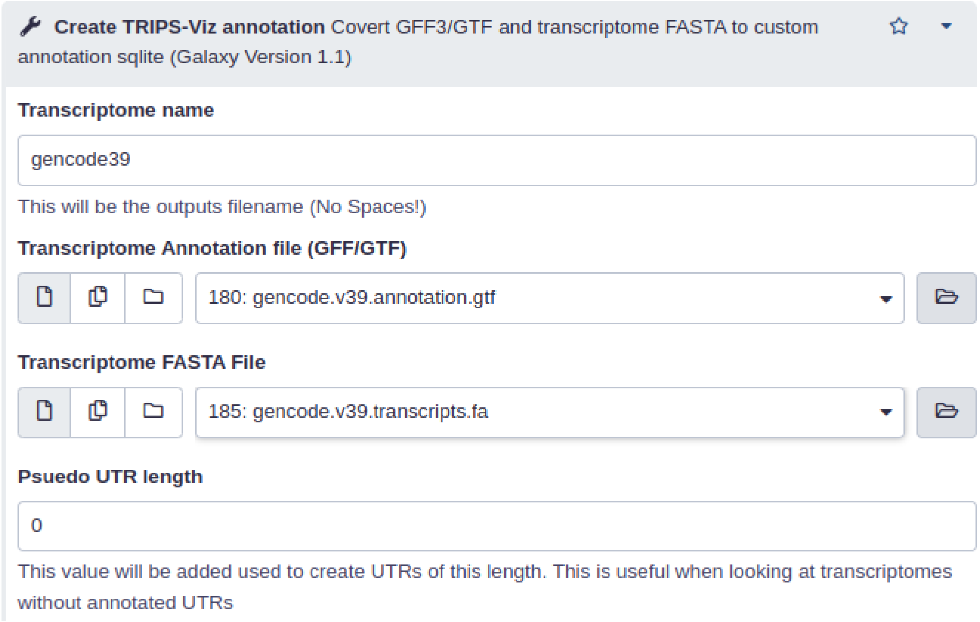
Name	Size	Type	Genome	Settings	Status
sample_human_1mli	351.4 MB	fastqsa...	Human Dec. 20...	⚙	100% ✓
gencode.v39.transcr	426.8 MB	fasta	Human Dec. 20...	⚙	100% ✓
gencode.v39.annota	1.4 GB	gtf	Human Dec. 20...	⚙	66%

Then you need to use following input files and parameters:

- Transcriptome name: '*gencode39*'
- Transcriptome annotation file: uploaded gtf
- Transcriptome fasta file: uploaded fasta
- Pseudo UTR length: 0
- Transcript id: *ENST00000456328.2*
- Gene name: *DDX11L1*

Gene and transcript id are taken from the beginning of the gtf file, it helps to better parse the gtf file since they typically have different formats depending on a gene annotation provider and organisms. Pseudo UTR length is not set on 0 when the transcriptome does not have UTRs (like yeast transcriptomes, in this case typically you would choose 300). Building annotation sqlite may take a while, so we provide it

in a dropbox:
https://www.dropbox.com/s/pxita1ebiosbrz5/Galaxy_G39_ann.sqlite?dl=0.
Push 'Execute'.



Create TRIPS-Viz annotation Covert GFF3/GTF and transcriptome FASTA to custom annotation sqlite (Galaxy Version 1.1)

Transcriptome name

gencode39

This will be the outputs filename (No Spaces!)

Transcriptome Annotation file (GFF/GTF)

180: gencode.v39.annotation.gtf

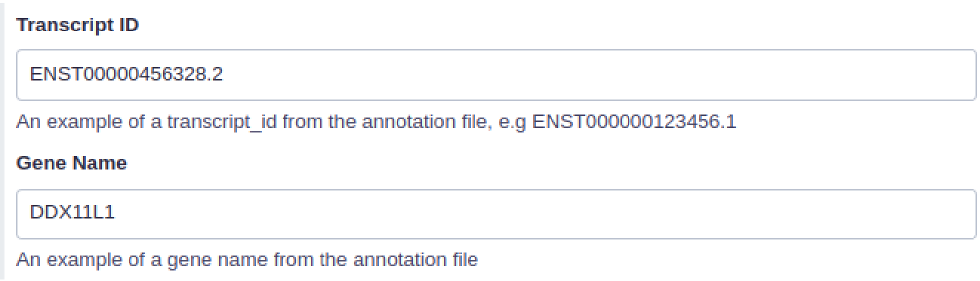
Transcriptome FASTA File

185: gencode.v39.transcripts.fa

Psuedo UTR length

0

This value will be added used to create UTRs of this length. This is useful when looking at transcriptomes without annotated UTRs



Transcript ID

ENST00000456328.2

An example of a transcript_id from the annotation file, e.g ENST000000123456.1

Gene Name

DDX11L1

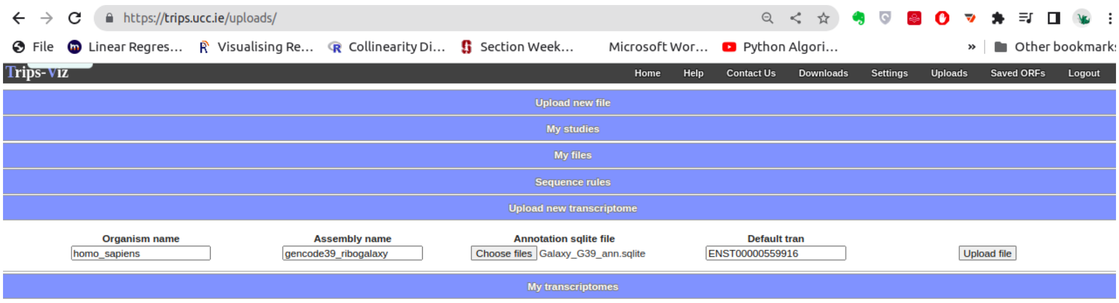
An example of a gene name from the annotation file

This way you can create sqlite annotation files for organisms or releases that are not available as built-ins.

6. We can download sqlite files from step 5 and upload it to the Trips-viz. In addition we will need GENCODE v39 annotation in sqlite which we can be downloaded from here: https://www.dropbox.com/s/pxita1ebiosbrz5/Galaxy_G39_ann.sqlite?dl=0 (or create on step 5').

How to upload annotation sqlite file: go to https://trips.ucc.ie/uploads/ and choose Upload new transcriptome.

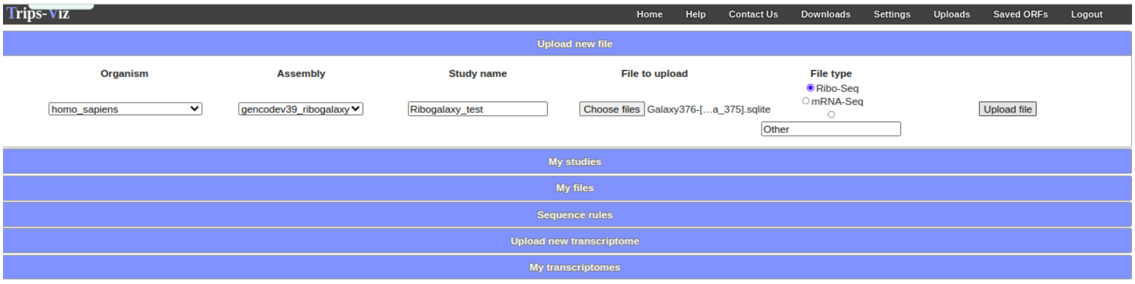
- Organism name: '*homo_sapiens*'
- Assembly name: '*gencode39_ribogalaxy'* (though you can put any name of assembly and organism)
- Default transcript: *ENST00000559916*
- Choose annotation sqlite file for uploading (it's ~3Gb)



Organism name	Assembly name	Annotation sqlite file	Default tran
homo_sapiens	gencode39_ribogalaxy	Galaxy_G39_ann.sqlite	ENST00000559916

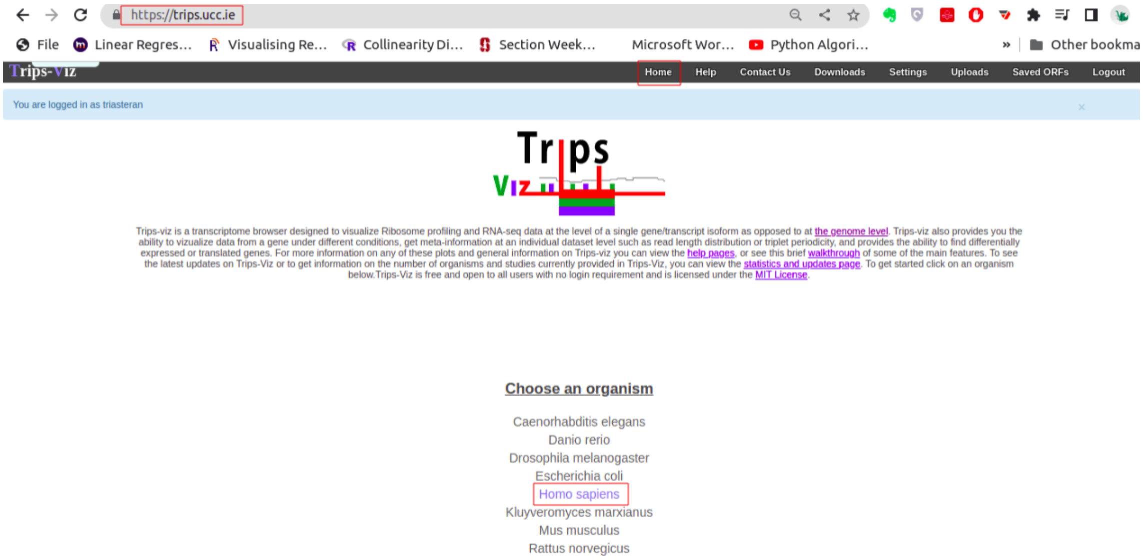
Next, choose Upload new file to upload the sample bam sqlite.

- Organism: homo_sapiens
- Assembly: gencode39_ribogalaxy
- choose sqlite file for uploading.



Organism	Assembly	Study name	File to upload	File type
homo_sapiens	gencodev39_ribogalaxy	Ribogalaxy_test	Galaxy376-[...a_375].sqlite	● Ribo-Seq ○ mRNA-Seq ○ Other

Then you can return to the homepage of Trips-viz, choose *Homo sapiens*, choose *gencode39_ribogalaxy* as transcriptome and choose a single transcript plot.

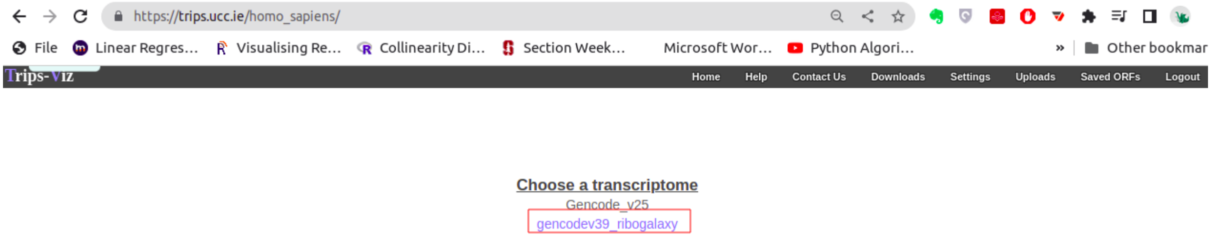


You are logged in as triasteran

Trips-viz is a transcriptome browser designed to visualize Ribosome profiling and RNA-seq data at the level of a single gene/transcript isoform as opposed to at the genome level. Trips-viz also provides you the ability to vizualize data from a gene under different conditions, get meta-information at an individual dataset level such as read length distribution or triplet periodicity, and provides the ability to find differentially expressed or translated genes. For more information on any of these plots and general information on Trips-viz you can view the help pages, or see this brief walkthrough of some of the main features. To see the latest updates on Trips-Viz or to get information on the number of organisms and studies currently provided in Trips-Viz, you can view the statistics and updates page. To get started click on an organism below.Trips-Viz is free and open to all users with no login requirement and is licensed under the MIT License.

Choose an organism

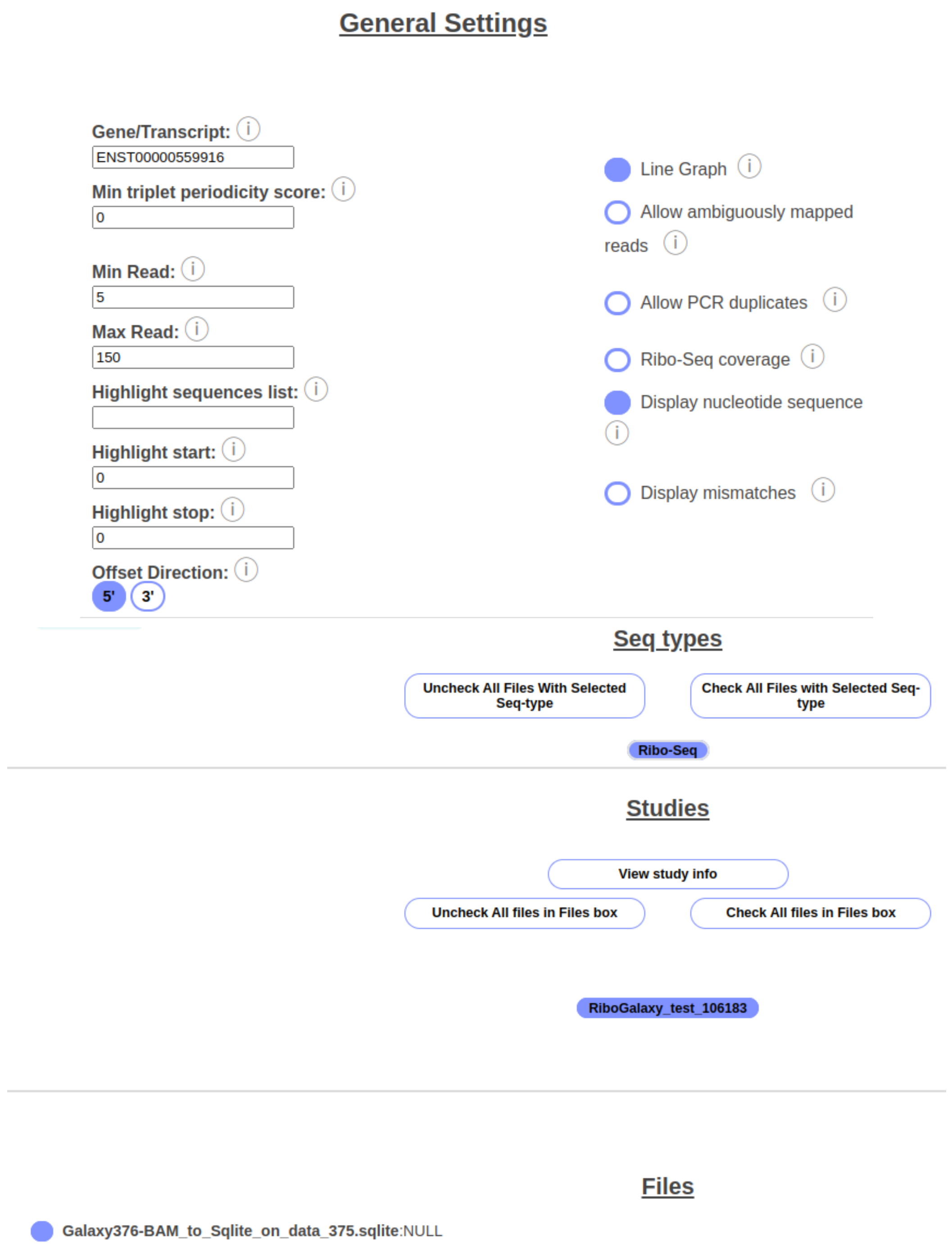
Caenorhabditis elegans
Danio rerio
Drosophila melanogaster
Escherichia coli
Homo sapiens
Kluyveromyces marxianus
Mus musculus
Rattus norvegicus



Choose a transcriptome

Gencode_v25
gencodev39_ribogalaxy

Originally we selected reads that map on a set of 12 transcripts. Now you can explore their subcodon profiles, e.g. by selecting the '**Single transcript plot**' and supplying a transcript: *ENST00000559916*:



General Settings

Gene/Transcript: ⓘ
ENST00000559916

Min triplet periodicity score: ⓘ
0

Min Read: ⓘ
5

Max Read: ⓘ
150

Highlight sequences list: ⓘ

Highlight start: ⓘ
0

Highlight stop: ⓘ
0

Offset Direction: ⓘ
● 5' ○ 3'

● Line Graph ⓘ

○ Allow ambiguously mapped reads ⓘ

○ Allow PCR duplicates ⓘ

○ Ribo-Seq coverage ⓘ

● Display nucleotide sequence ⓘ

○ Display mismatches ⓘ

Seq types

Uncheck All Files With Selected Seq-type
Check All Files with Selected Seq-type

● Ribo-Seq

Studies

View study info
Uncheck All files in Files box
Check All files in Files box

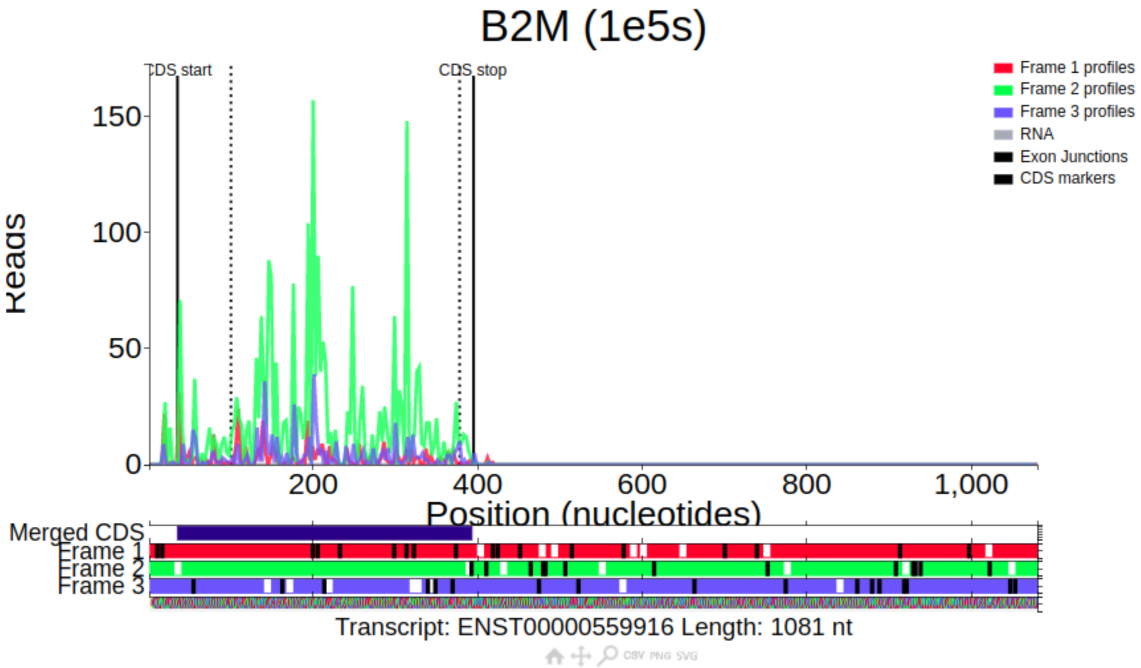
● RiboGalaxy_test_106183

Files

● Galaxy376-BAM_to_Sqlite_on_data_375.sqlite:NULL

View Plot

Direct link to this plot



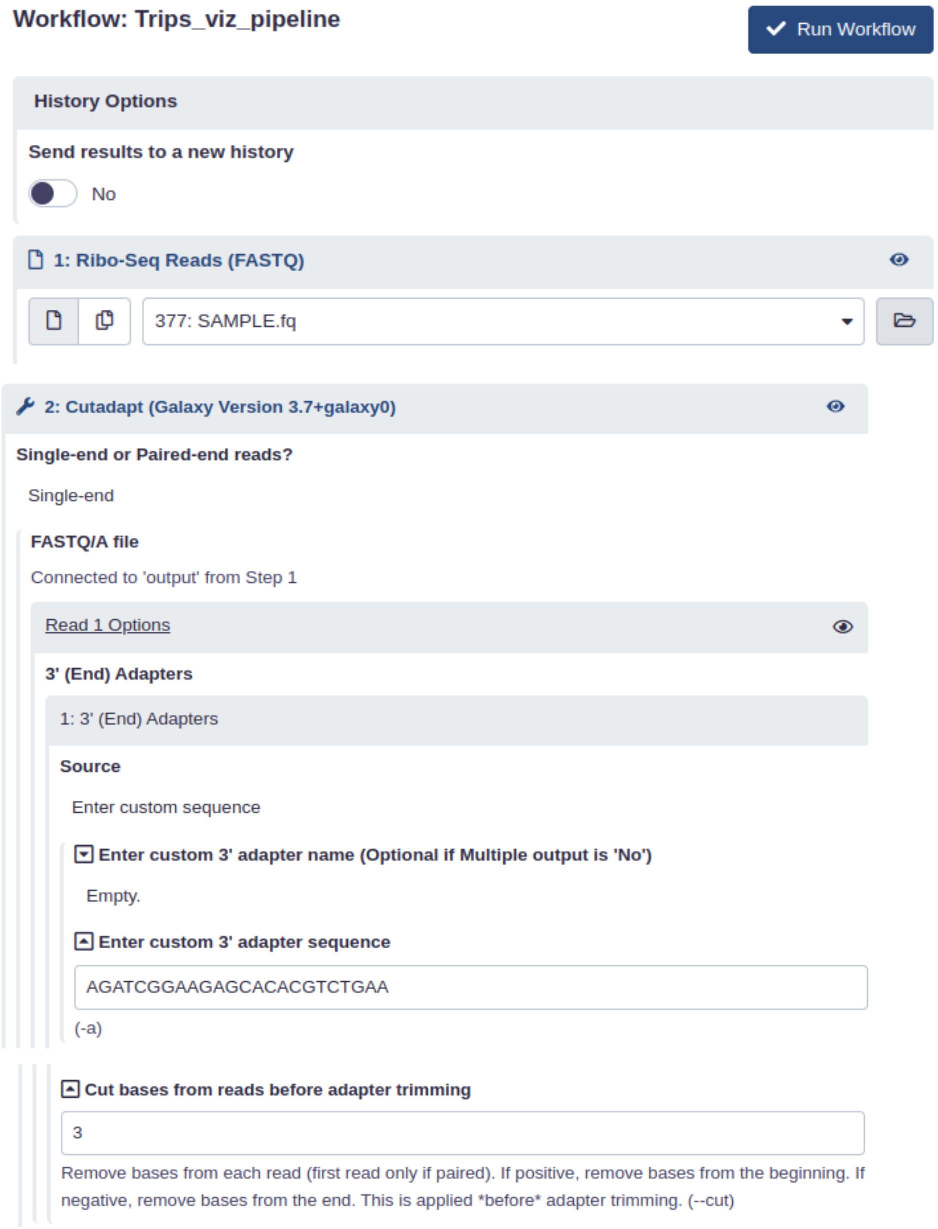
The list of all transcripts:

ENST00000559916,ENST00000371222,ENST00000621592,ENST00000303004,ENST00000303004,ENST00000366794,ENST00000327443,ENST00000575671,ENST00000370339,ENST00000700002,ENST00000306077,ENST00000367975.

STAGE IIb: mapping on a transcriptome using a Workflow.

Let's use a workflow instead of doing things step-by-step. Of note, this workflow is not tailored to any possible read structure, e.g. in the present workflow there is an additional cutadapt step for removing barcodes. In case you don't need it, you can create your own workflow (or edit this one locally) based on the provided one.

First, you need to access it via: Shared data -> Workflows -> Trips_viz_pipeline (author triasteran). Click on 'run' button at the top right corner. Now you need to specify all the parameters from the STAGE IIa. You may notice that all of them are already set and you only need to choose the input file (SAMPLE.fq).



Workflow: Trips_viz_pipeline

History Options

Send results to a new history

○ No

1: Ribo-Seq Reads (FASTQ)

377: SAMPLE.fq

2: Cutadapt (Galaxy Version 3.7+galaxy0)

Single-end or Paired-end reads?

Single-end

FASTQ/A file

Connected to 'output' from Step 1

Read 1 Options

3' (End) Adapters

1: 3' (End) Adapters

Source

Enter custom sequence

Enter custom 3' adapter name (Optional if Multiple output is 'No')

Empty.

Enter custom 3' adapter sequence

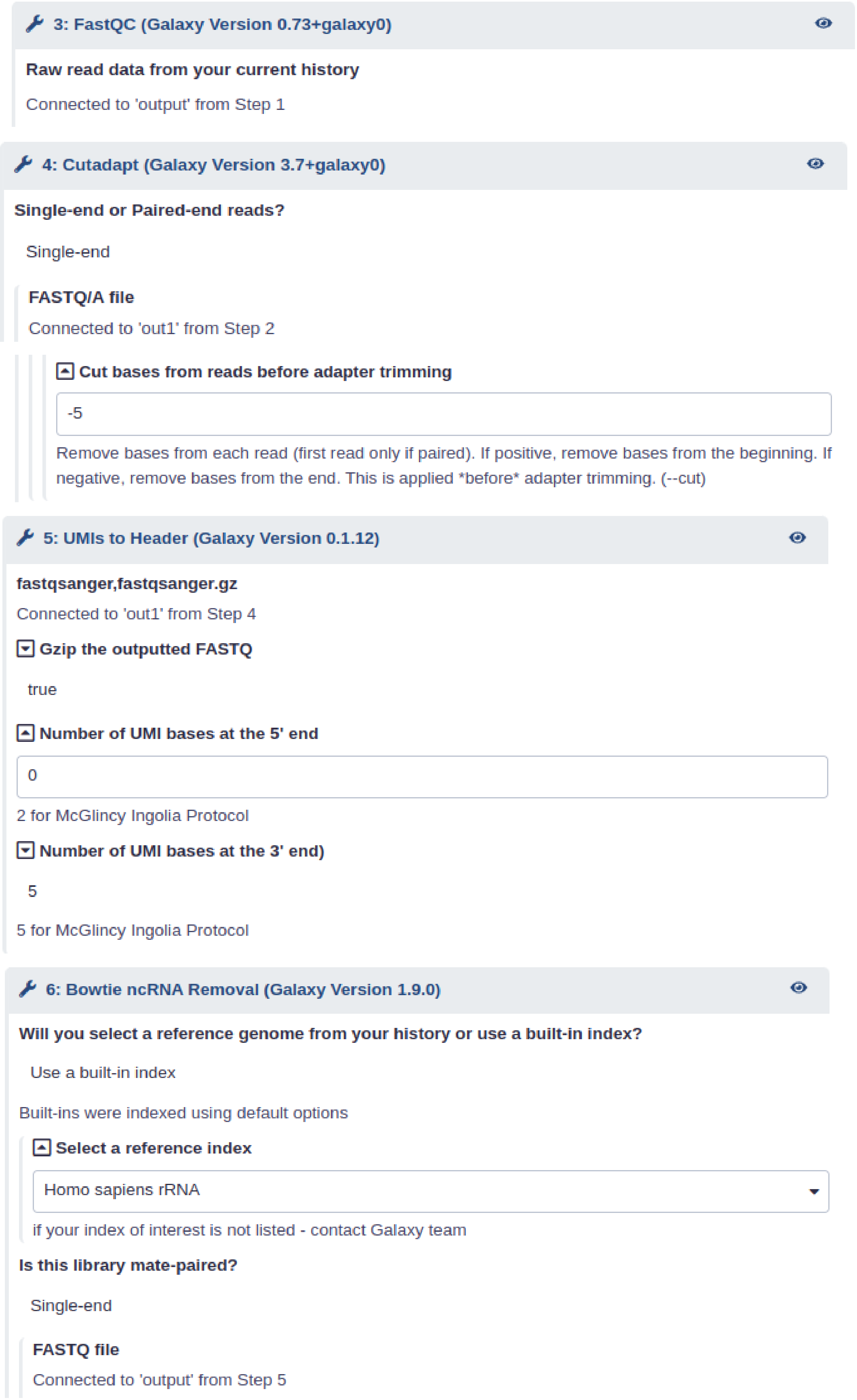
AGATCGGAAGAGCACACGTCTGAA

(-a)

Cut bases from reads before adapter trimming

3

Remove bases from each read (first read only if paired). If positive, remove bases from the beginning. If negative, remove bases from the end. This is applied *before* adapter trimming. (--cut)



3: FastQC (Galaxy Version 0.73+galaxy0)

Raw read data from your current history

Connected to 'output' from Step 1

4: Cutadapt (Galaxy Version 3.7+galaxy0)

Single-end or Paired-end reads?

Single-end

FASTQ/A file

Connected to 'out1' from Step 2

Cut bases from reads before adapter trimming

-5

Remove bases from each read (first read only if paired). If positive, remove bases from the beginning. If negative, remove bases from the end. This is applied *before* adapter trimming. (--cut)

5: UMIs to Header (Galaxy Version 0.1.12)

fastqsanger,fastqsanger.gz

Connected to 'out1' from Step 4

Gzip the outputted FASTQ

true

Number of UMI bases at the 5' end

0

2 for McGlincy Ingolia Protocol

Number of UMI bases at the 3' end)

5

5 for McGlincy Ingolia Protocol

6: Bowtie ncRNA Removal (Galaxy Version 1.9.0)

Will you select a reference genome from your history or use a built-in index?

Use a built-in index

Built-ins were indexed using default options

Select a reference index

Homo sapiens rRNA

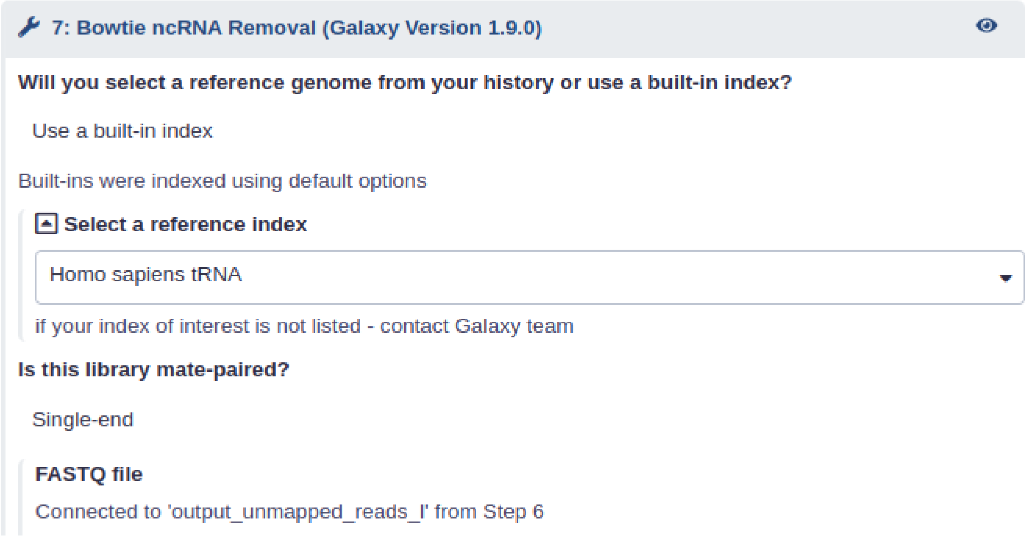
if your index of interest is not listed - contact Galaxy team

Is this library mate-paired?

Single-end

FASTQ file

Connected to 'output' from Step 5



7: Bowtie ncRNA Removal (Galaxy Version 1.9.0)

Will you select a reference genome from your history or use a built-in index?

Use a built-in index

Built-ins were indexed using default options

Select a reference index

Homo sapiens tRNA

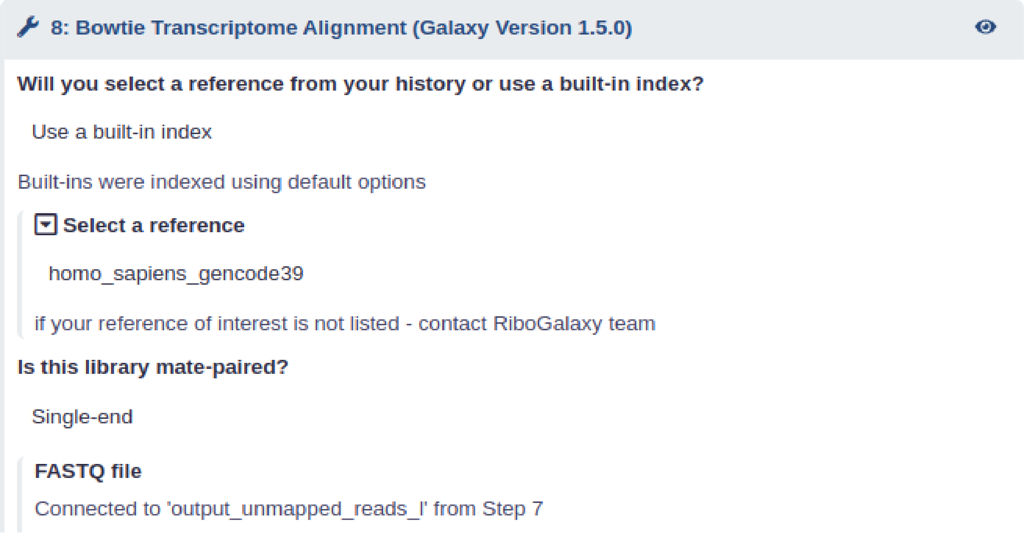
if your index of interest is not listed - contact Galaxy team

Is this library mate-paired?

Single-end

FASTQ file

Connected to 'output_unmapped_reads_l' from Step 6



8: Bowtie Transcriptome Alignment (Galaxy Version 1.5.0)

Will you select a reference from your history or use a built-in index?

Use a built-in index

Built-ins were indexed using default options

Select a reference

homo_sapiens_gencode39

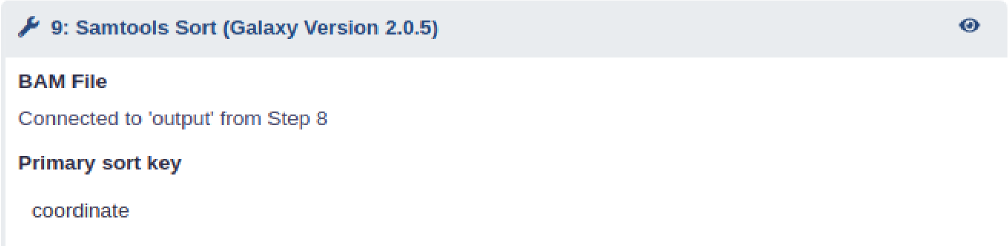
if your reference of interest is not listed - contact RiboGalaxy team

Is this library mate-paired?

Single-end

FASTQ file

Connected to 'output_unmapped_reads_l' from Step 7



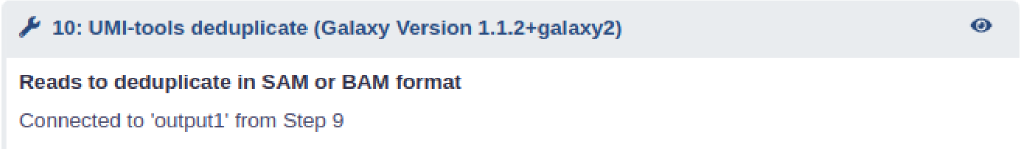
9: Samtools Sort (Galaxy Version 2.0.5)

BAM File

Connected to 'output' from Step 8

Primary sort key

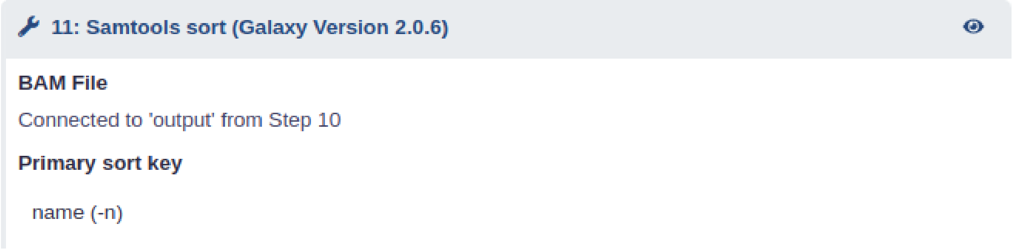
coordinate



10: UMI-tools deduplicate (Galaxy Version 1.1.2+galaxy2)

Reads to deduplicate in SAM or BAM format

Connected to 'output1' from Step 9



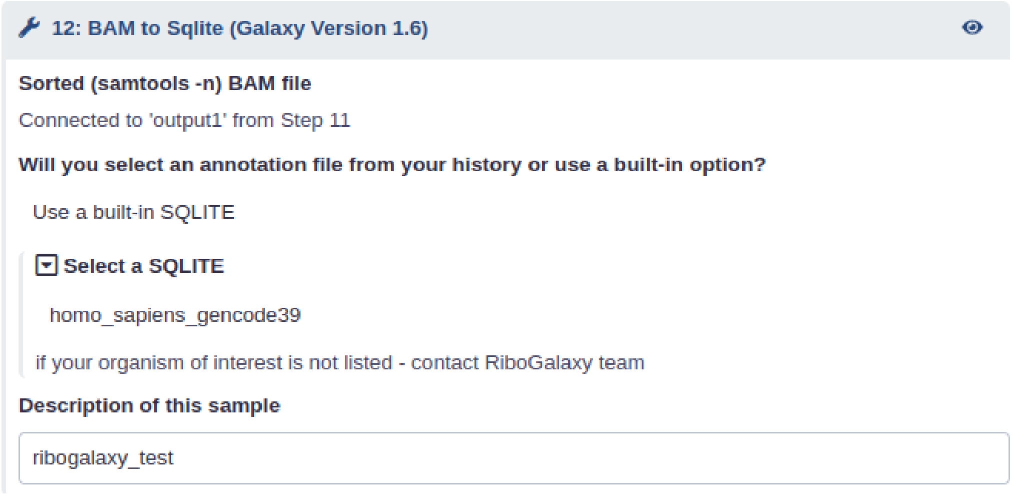
11: Samtools sort (Galaxy Version 2.0.6)

BAM File

Connected to 'output' from Step 10

Primary sort key

name (-n)



12: BAM to Sqlite (Galaxy Version 1.6)

Sorted (samtools -n) BAM file

Connected to 'output1' from Step 11

Will you select an annotation file from your history or use a built-in option?

Use a built-in SQLITE

Select a SQLITE

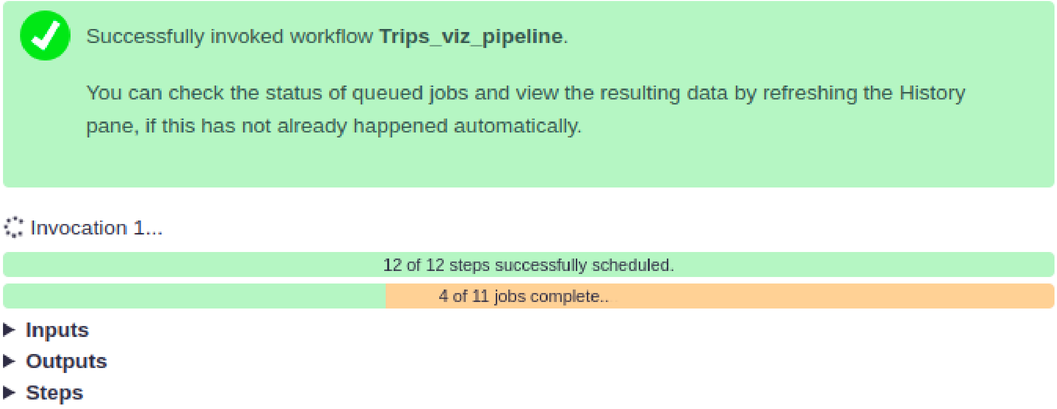
homo_sapiens_gencode39

if your organism of interest is not listed - contact RiboGalaxy team

Description of this sample

ribogalaxy_test

Once all parameters are set, click on ‘Run Workflow’ on top of the page.



Successfully invoked workflow **Trips_viz_pipeline**.

You can check the status of queued jobs and view the resulting data by refreshing the History pane, if this has not already happened automatically.

Invocation 1...

12 of 12 steps successfully scheduled.

4 of 11 jobs complete..

▶ Inputs

▶ Outputs

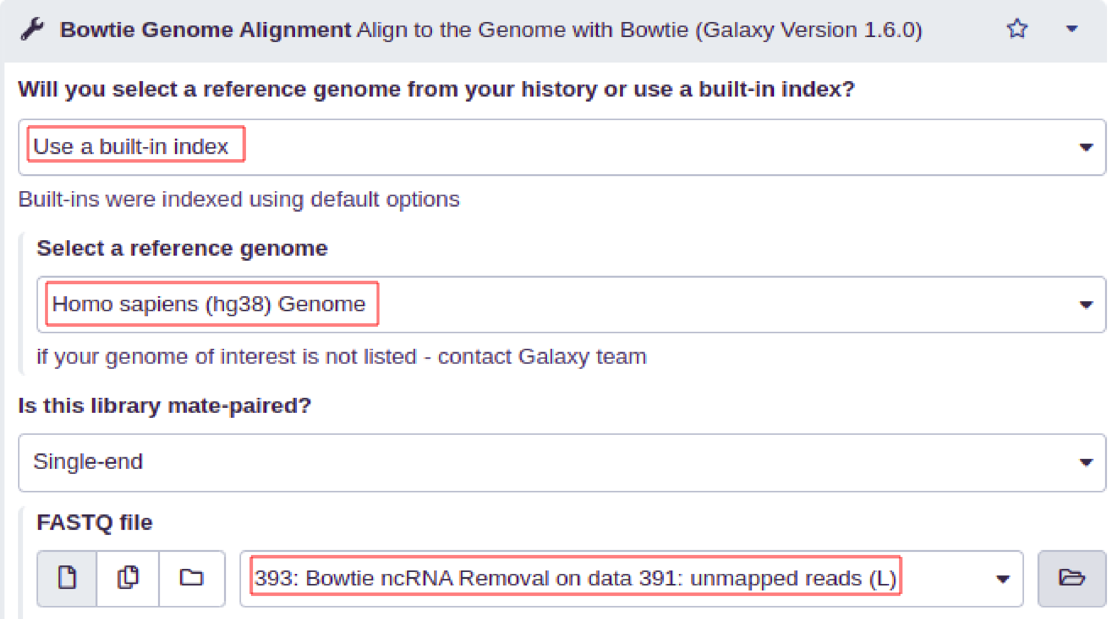
▶ Steps

STAGE IIIa: mapping on a genome.

Let's briefly go through the main steps.

1. First we will map reads after STAGE I (ribosome protected fragments) on the genome using ***Bowtie Genome Alignment***.
2. Then we sort alignments based on coordinates using ***Samtools sort***.
3. Next we will deduplicate a sample based on UMIs using ***UMI-tools deduplicate***.
4. Then we again sort alignments based on coordinates using Samtools sort.
5. We need to obtain chromosome sizes by using the Get ***Chromosome Sizes tool***.
6. Using the deduplicated bam file from step 4, now we can create ribosome profile in bed format using ***Create Ribosome Profiles*** tool.
7. Next we will convert the bed file to BigWig so that it can be uploaded and visualised in GWIPS-viz by using ***Convert a BED File to a BigWig*** tool*.*
8. In order to upload the resulting BigWig track file to GWIPS-viz, we will need to create a link using the Generate **Custom Track** tool.

1. Map reads (RPFs) that were cleaned from rRNA/tRNA contamination (after the last step of STAGE I) on the genome using ***Bowtie Genome Alignment*** in **GWIPS-Viz (genomic alignment) branch**. You can choose the built-in index *Homo sapiens (hg38) Genome*. Push 'Execute'.



Bowtie Genome Alignment Align to the Genome with Bowtie (Galaxy Version 1.6.0)

Will you select a reference genome from your history or use a built-in index?

Use a built-in index

Built-ins were indexed using default options

Select a reference genome

Homo sapiens (hg38) Genome

if your genome of interest is not listed - contact Galaxy team

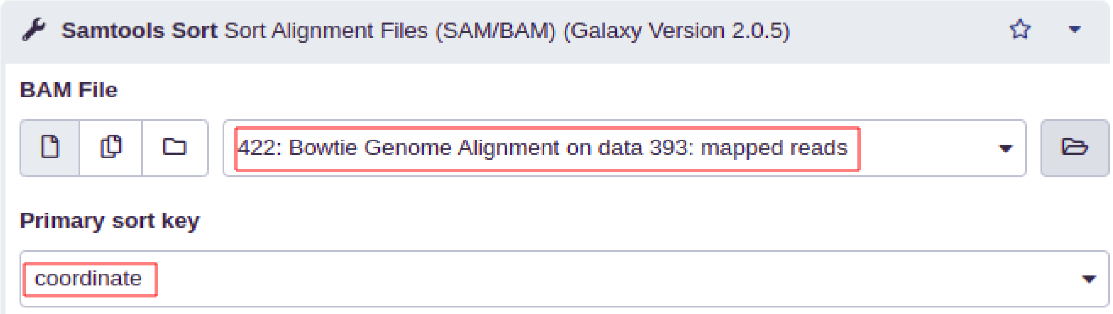
Is this library mate-paired?

Single-end

FASTQ file

393: Bowtie ncRNA Removal on data 391: unmapped reads (L)

2. Sort the resulting bam file by coordinate using ***Samtools sort*** from **GWIPS-Viz (genomic alignment) branch**. Push 'Execute'.



Samtools Sort Sort Alignment Files (SAM/BAM) (Galaxy Version 2.0.5)

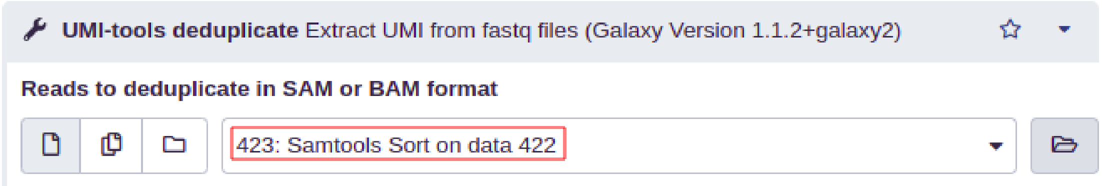
BAM File

422: Bowtie Genome Alignment on data 393: mapped reads

Primary sort key

coordinate

3. This step can be performed if you want to deduplicate your sample (remove PCR duplicates) using UMIs. It can be done with ***UMI-tools deduplicate*** from **UMI and barcodes**. Input file is bam sorted by coordinate.You can also choose to output log file. Push 'Execute'.

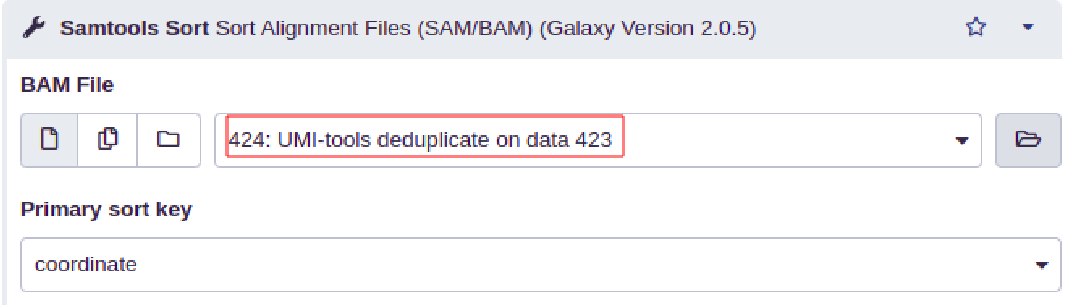


UMI-tools deduplicate Extract UMI from fastq files (Galaxy Version 1.1.2+galaxy2)

Reads to deduplicate in SAM or BAM format

423: Samtools Sort on data 422

4. Sort the resulting deduplicated bam file by coordinate using ***Samtools sort*** from **GWIPS-Viz (genomic alignment) branch**. Push 'Execute'.



Samtools Sort Sort Alignment Files (SAM/BAM) (Galaxy Version 2.0.5)

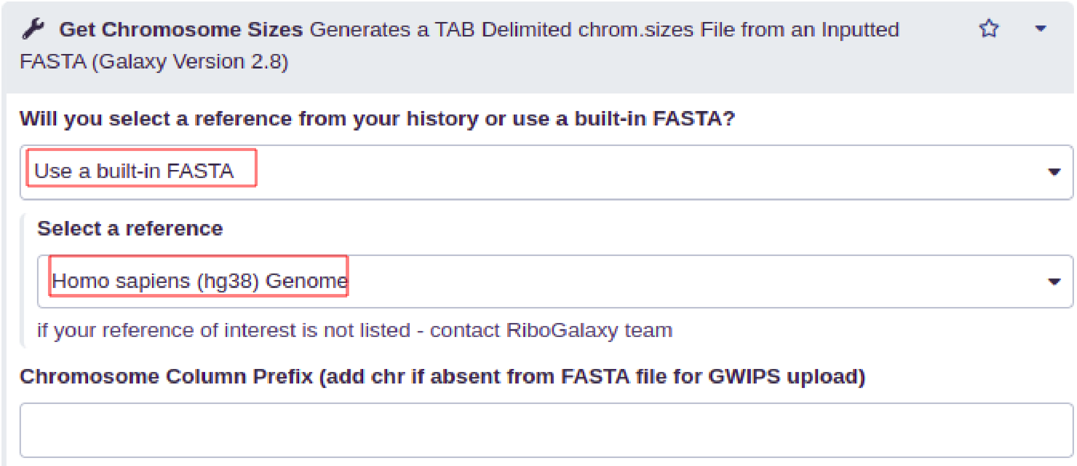
BAM File

424: UMI-tools deduplicate on data 423

Primary sort key

coordinate

5. Now we will get chromosome sizes for subsequent transformation of alignment to the ribosome profile. We will use the ***Get Chromosome Sizes*** tool from **GWIPS-Viz (genomic alignment) branch**. Select built-in fasta, Homo sapiens (hg38) Genome. Push 'Execute'.



Get Chromosome Sizes Generates a TAB Delimited chrom.sizes File from an Inputted FASTA (Galaxy Version 2.8)

Will you select a reference from your history or use a built-in FASTA?

Use a built-in FASTA

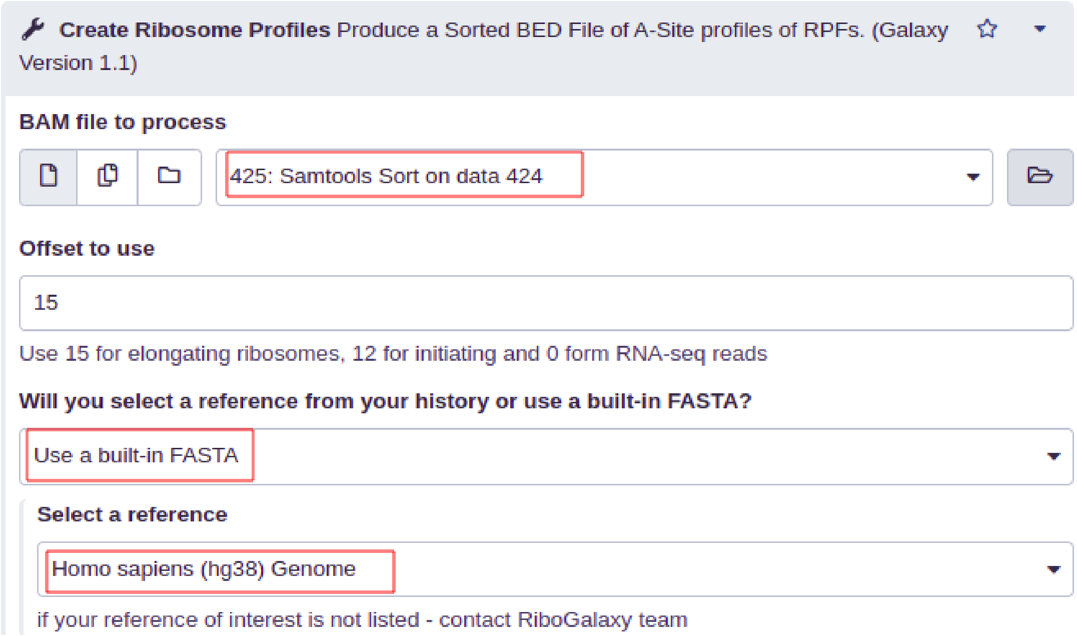
Select a reference

Homo sapiens (hg38) Genome

if your reference of interest is not listed - contact RiboGalaxy team

Chromosome Column Prefix (add chr if absent from FASTA file for GWIPS upload)

6. Create ribosome profile in bed format from sorted by coordinate deduplicated bam file from step 4 using ***Create Ribosome Profiles*** tool in **GWIPS-Viz (genomic alignment) branch**. Choose built-in fasta, *Homo sapiens (hg38) Genome*. Push 'Execute'.



Create Ribosome Profiles Produce a Sorted BED File of A-Site profiles of RPFs. (Galaxy Version 1.1)

BAM file to process

425: Samtools Sort on data 424

Offset to use

15

Use 15 for elongating ribosomes, 12 for initiating and 0 form RNA-seq reads

Will you select a reference from your history or use a built-in FASTA?

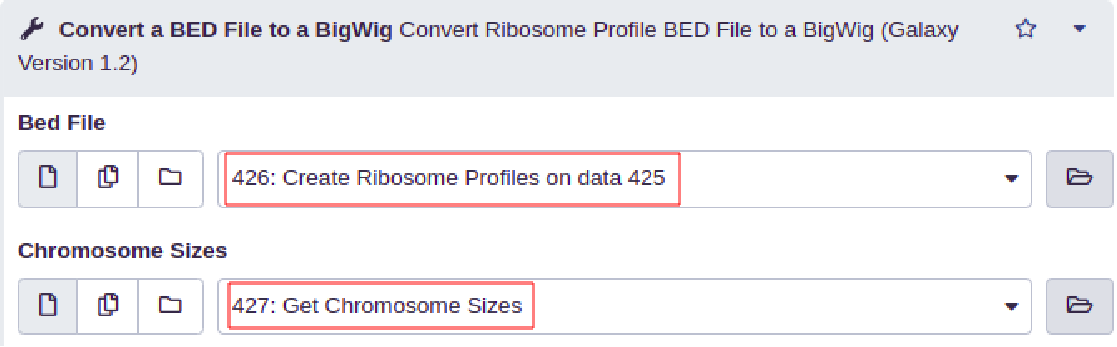
Use a built-in FASTA

Select a reference

Homo sapiens (hg38) Genome

if your reference of interest is not listed - contact RiboGalaxy team

7. Next we need to convert the bed file from step 6 to BigWig file. BigWig can be uploaded to genome browsers, e.g. GWIPS-viz. We will use the ***Convert a BED File to a BigWig*** tool from **GWIPS-Viz (genomic alignment) branch**. Another input is chromosome sizes file from step 3. Push 'Execute'.



Convert a BED File to a BigWig Convert Ribosome Profile BED File to a BigWig (Galaxy Version 1.2)

Bed File

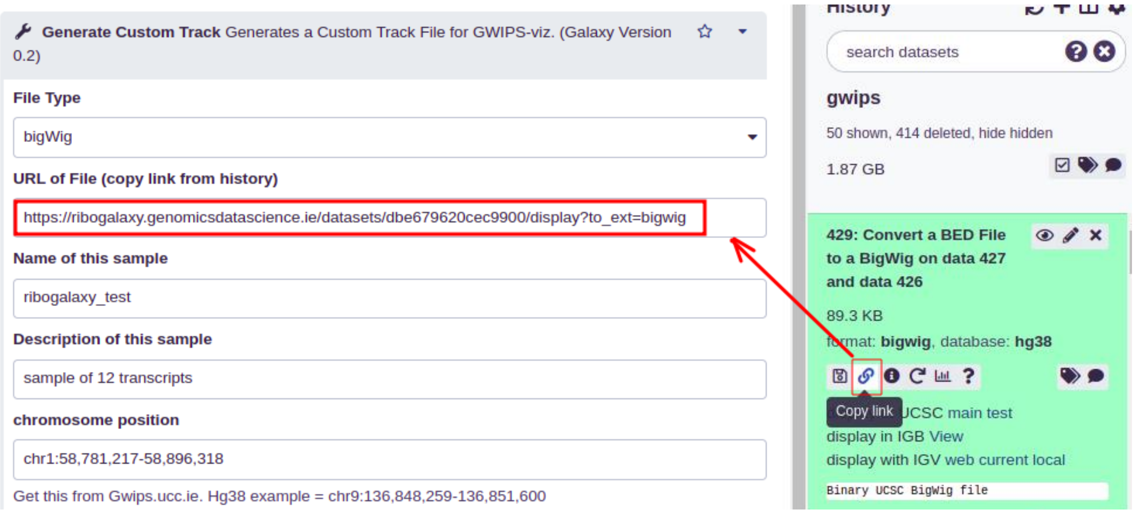
426: Create Ribosome Profiles on data 425

Chromosome Sizes

427: Get Chromosome Sizes

8. Finally, we can generate the file for easy upload of the BigWig file generated on step 7 to the GWIPS-viz browser. We will use **Generate Custom Track** from **GWIPS-Viz (genomic alignment) branch**. You will need to copy link to the BigWig file and use it as input:

Add name and description of the sample, as well as any chromosome position of interest, e.g. `chr1:58,781,217-58,896,318` (it can be changed in genome browser). Push 'Execute'. This tool will output a file containing a link. You need to download this file and then upload to GWIPS-viz browser.



Generate Custom Track Generates a Custom Track File for GWIPS-viz. (Galaxy Version 0.2)

File Type

bigWig

URL of File (copy link from history)

https://ribogalaxy.genomicsdatascience.ie/datasets/dbe679620cec9900/display?to_ext=bigwig

Name of this sample

ribogalaxy_test

Description of this sample

sample of 12 transcripts

chromosome position

chr1:58,781,217-58,896,318

Get this from Gwips.ucc.ie. Hg38 example = chr9:136,848,259-136,851,600

History

search datasets

gwips

50 shown, 414 deleted, hide hidden

1.87 GB

429: Convert a BED File to a BigWig on data 427 and data 426

89.3 KB

format: **bigwig**, database: **hg38**

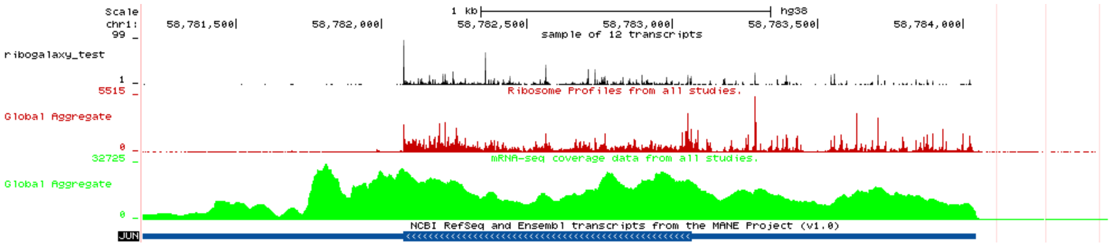
display at UCSC main test

display in IGB View

display with IGV web current local

Binary UCSC BigWig file

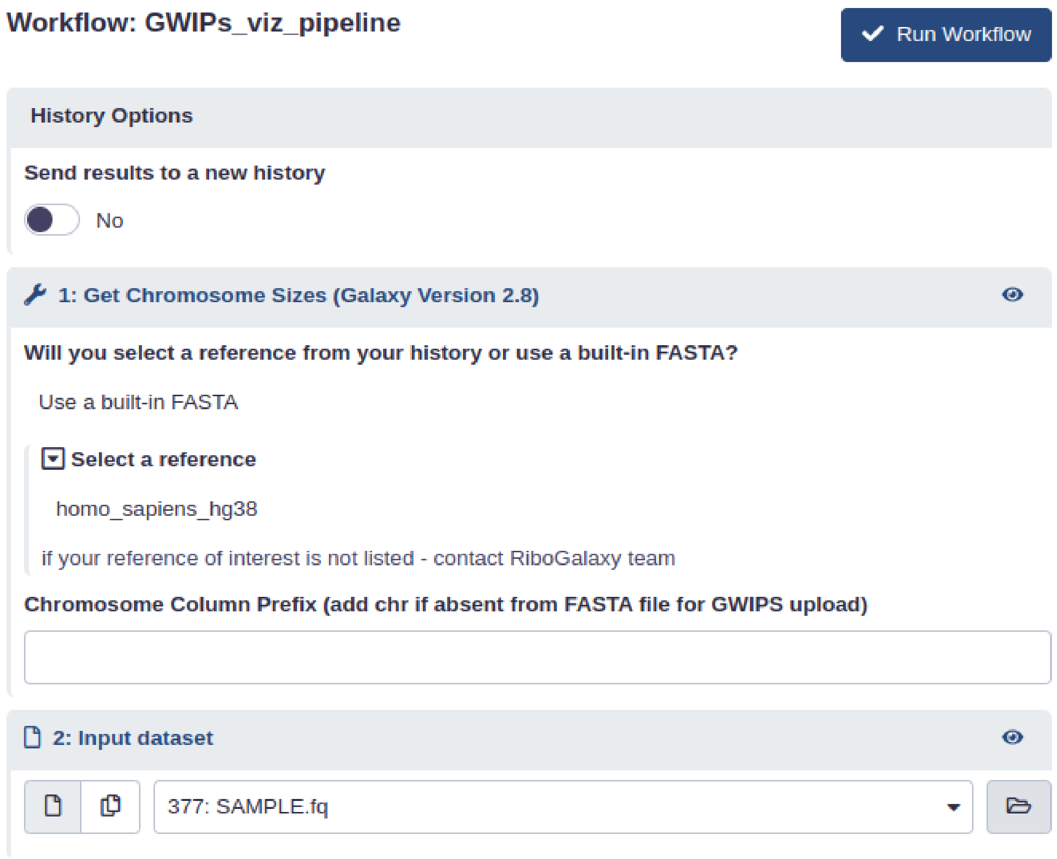
In order to upload this file, go to https://gwips.ucc.ie/cgi-bin/hgGateway, select **My Data**, **Custom Tracks** and **add custom track. Choose file** and **submit**. You then can take a look at the data:



STAGE IIIb: mapping on a genome using Workflow.

Let's use a workflow instead of doing things step-by-step. Of note, this workflow is not tailored to any possible read structure, e.g. in the present workflow there is an additional cutadapt step for removing barcodes. In case you don't need it, you can create your own workflow (or edit this one locally) based on the provided one.

First, you need to access it via: Shared data -> Workflows -> GWIPs_viz_pipeline (author triasteran). Click on 'run' button at the top right corner. Now you need to specify all the parameters from the STAGE IIIa. Notice that all of them are already set except the input file which you need to specify: SAMPLE.fq.


Workflow: GWIPs_viz_pipeline

History Options

Send results to a new history

○ No

1: Get Chromosome Sizes (Galaxy Version 2.8)

Will you select a reference from your history or use a built-in FASTA?

Use a built-in FASTA

Select a reference

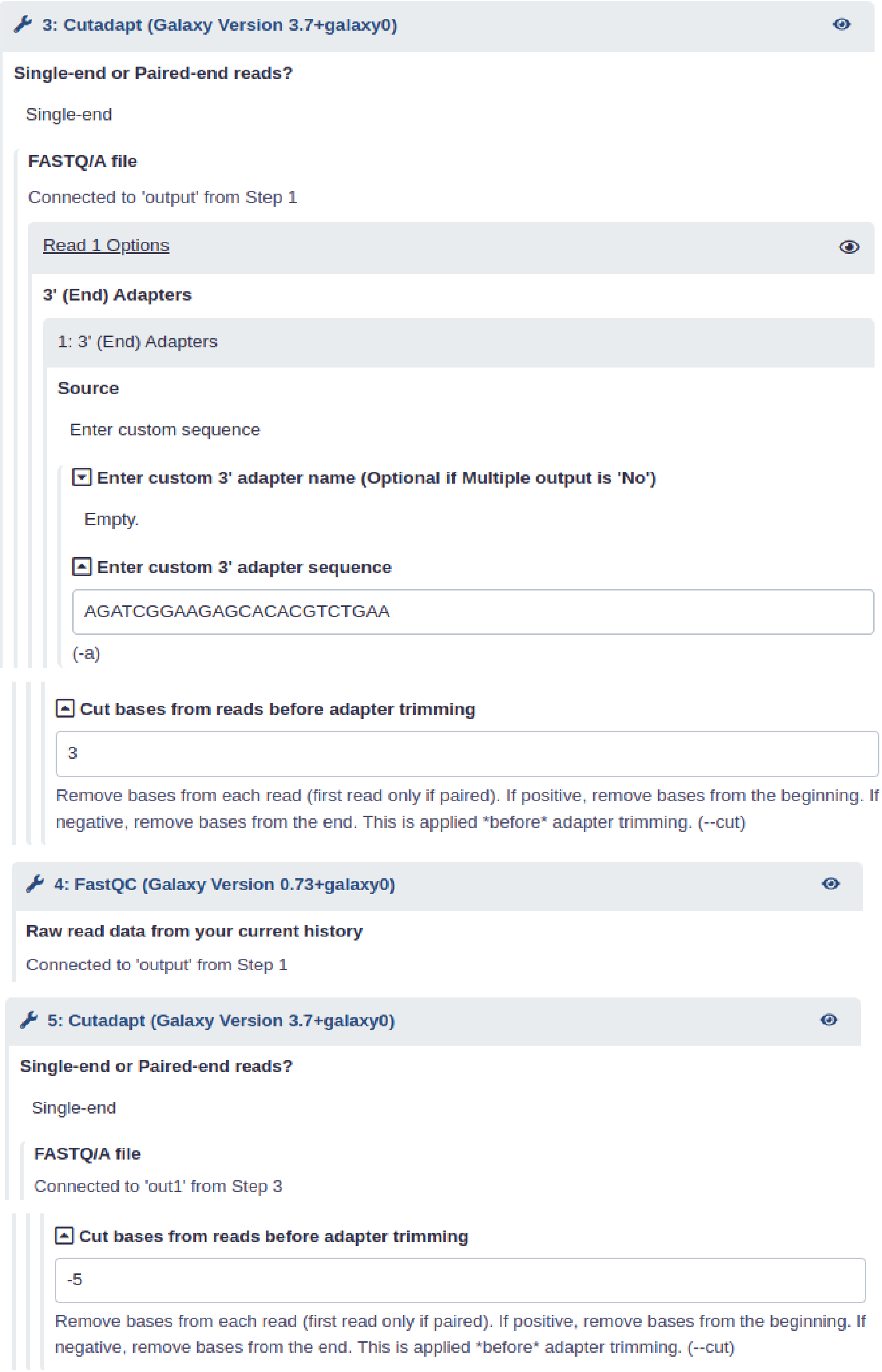
homo_sapiens_hg38

if your reference of interest is not listed - contact RiboGalaxy team

Chromosome Column Prefix (add chr if absent from FASTA file for GWIPS upload)

2: Input dataset

377: SAMPLE.fq



3: Cutadapt (Galaxy Version 3.7+galaxy0)

Single-end or Paired-end reads?

Single-end

FASTQ/A file

Connected to 'output' from Step 1

Read 1 Options

3' (End) Adapters

1: 3' (End) Adapters

Source

Enter custom sequence

Enter custom 3' adapter name (Optional if Multiple output is 'No')

Empty.

Enter custom 3' adapter sequence

AGATCGGAAGAGCACACGTCTGAA

(-a)

Cut bases from reads before adapter trimming

3

Remove bases from each read (first read only if paired). If positive, remove bases from the beginning. If negative, remove bases from the end. This is applied *before* adapter trimming. (--cut)

4: FastQC (Galaxy Version 0.73+galaxy0)

Raw read data from your current history

Connected to 'output' from Step 1

5: Cutadapt (Galaxy Version 3.7+galaxy0)

Single-end or Paired-end reads?

Single-end

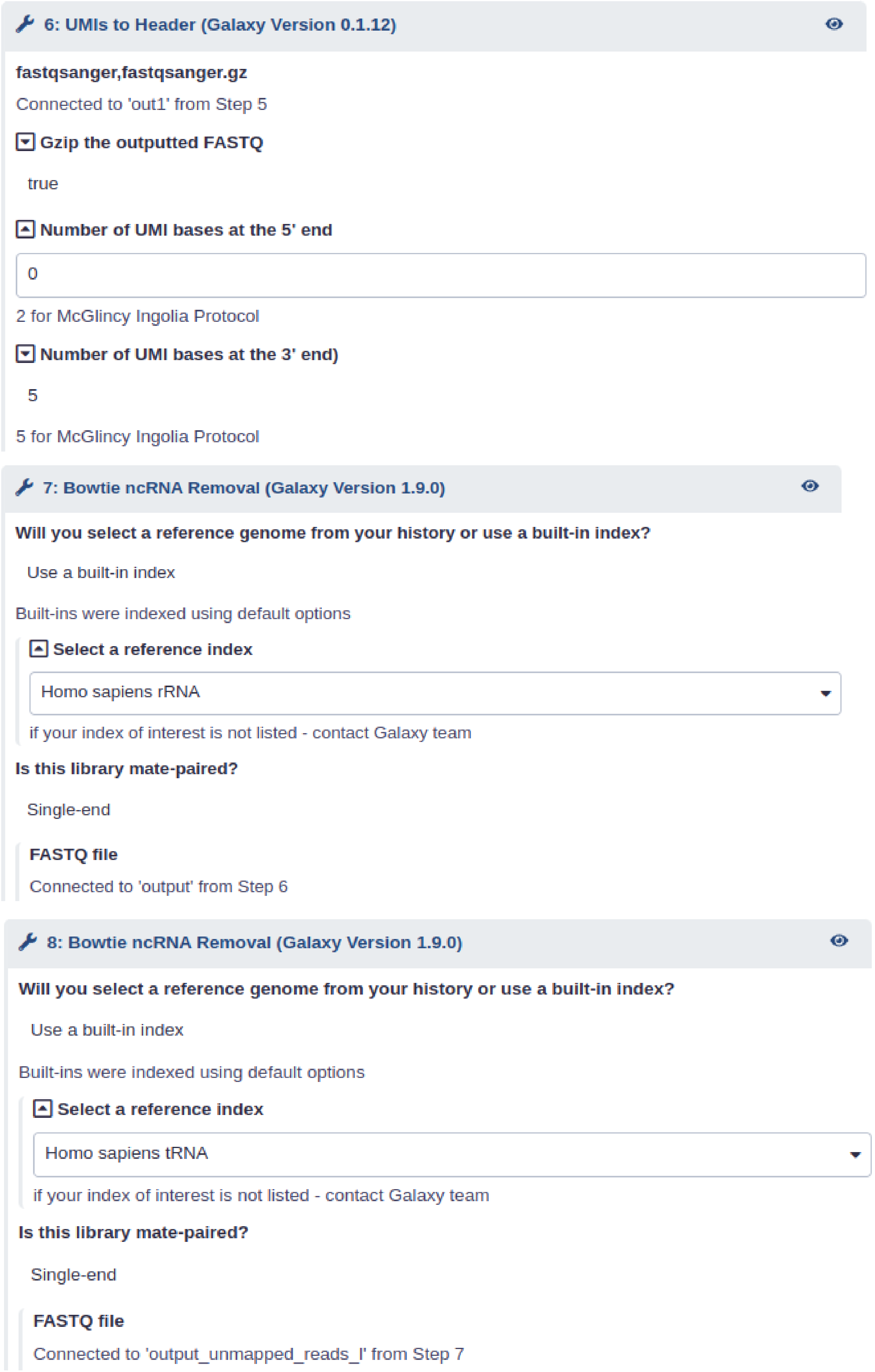
FASTQ/A file

Connected to 'out1' from Step 3

Cut bases from reads before adapter trimming

-5

Remove bases from each read (first read only if paired). If positive, remove bases from the beginning. If negative, remove bases from the end. This is applied *before* adapter trimming. (--cut)



6: UMIs to Header (Galaxy Version 0.1.12)

fastqsanger,fastqsanger.gz

Connected to 'out1' from Step 5

Gzip the outputted FASTQ

true

Number of UMI bases at the 5' end

0

2 for McGlincy Ingolia Protocol

Number of UMI bases at the 3' end)

5

5 for McGlincy Ingolia Protocol

7: Bowtie ncRNA Removal (Galaxy Version 1.9.0)

Will you select a reference genome from your history or use a built-in index?

Use a built-in index

Built-ins were indexed using default options

Select a reference index

Homo sapiens rRNA

if your index of interest is not listed - contact Galaxy team

Is this library mate-paired?

Single-end

FASTQ file

Connected to 'output' from Step 6

8: Bowtie ncRNA Removal (Galaxy Version 1.9.0)

Will you select a reference genome from your history or use a built-in index?

Use a built-in index

Built-ins were indexed using default options

Select a reference index

Homo sapiens tRNA

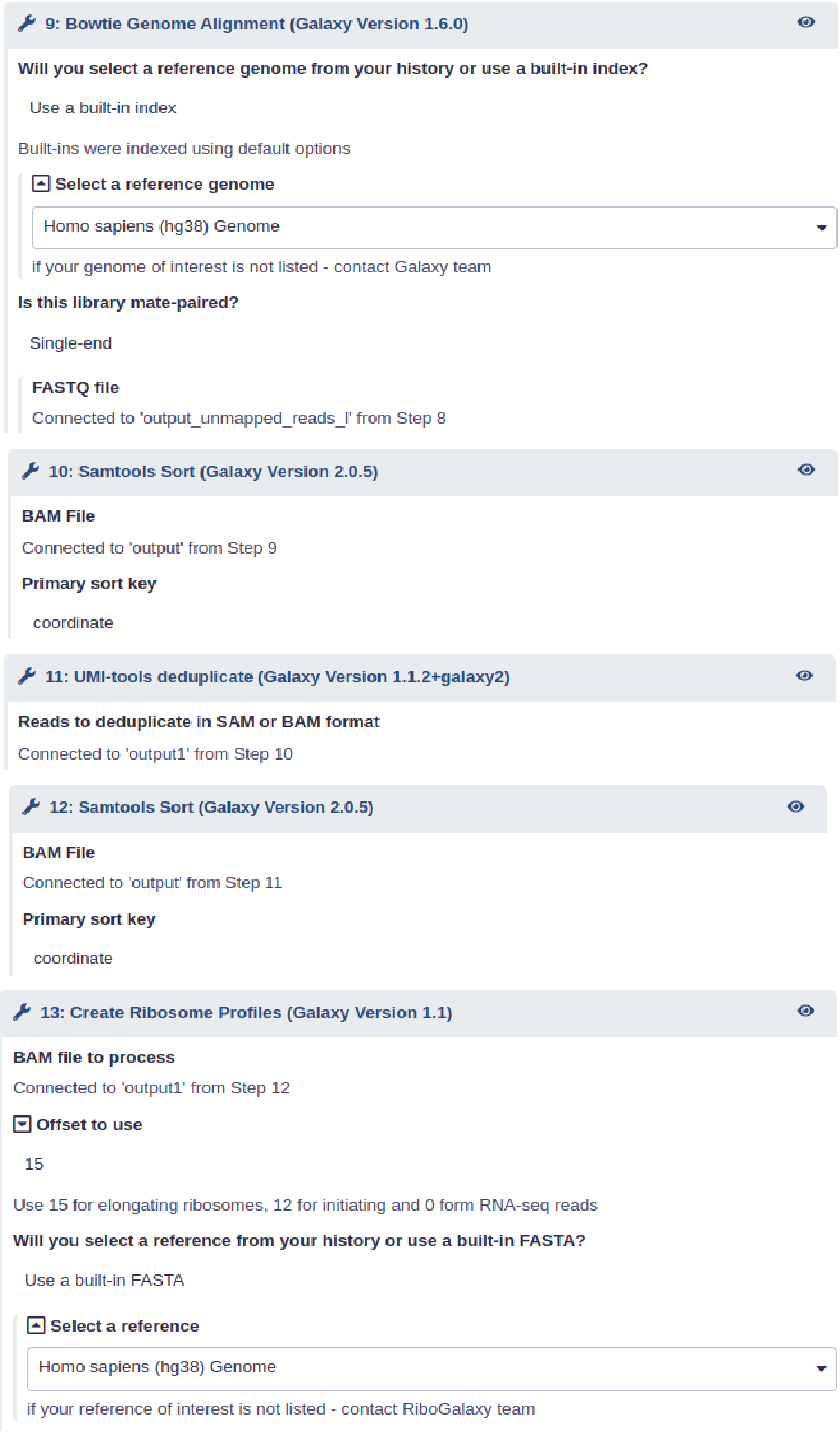
if your index of interest is not listed - contact Galaxy team

Is this library mate-paired?

Single-end

FASTQ file

Connected to 'output_unmapped_reads_l' from Step 7



9: Bowtie Genome Alignment (Galaxy Version 1.6.0)

Will you select a reference genome from your history or use a built-in index?

Use a built-in index

Built-ins were indexed using default options

Select a reference genome

Homo sapiens (hg38) Genome

if your genome of interest is not listed - contact Galaxy team

Is this library mate-paired?

Single-end

FASTQ file

Connected to 'output_unmapped_reads_l' from Step 8

10: Samtools Sort (Galaxy Version 2.0.5)

BAM File

Connected to 'output' from Step 9

Primary sort key

coordinate

11: UMI-tools deduplicate (Galaxy Version 1.1.2+galaxy2)

Reads to deduplicate in SAM or BAM format

Connected to 'output1' from Step 10

12: Samtools Sort (Galaxy Version 2.0.5)

BAM File

Connected to 'output' from Step 11

Primary sort key

coordinate

13: Create Ribosome Profiles (Galaxy Version 1.1)

BAM file to process

Connected to 'output1' from Step 12

Offset to use

15

Use 15 for elongating ribosomes, 12 for initiating and 0 form RNA-seq reads

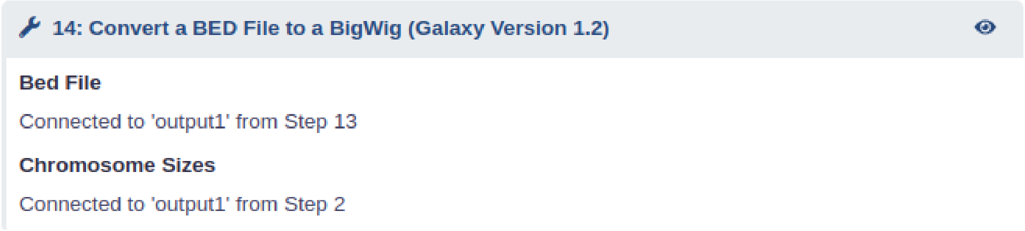
Will you select a reference from your history or use a built-in FASTA?

Use a built-in FASTA

Select a reference

Homo sapiens (hg38) Genome

if your reference of interest is not listed - contact RiboGalaxy team



14: Convert a BED File to a BigWig (Galaxy Version 1.2)

Bed File

Connected to 'output1' from Step 13

Chromosome Sizes

Connected to 'output1' from Step 2

Final step you have to do separately by using **Generate Custom Track** from **GWIPS-Viz (genomic alignment) branch** (the same way as described in STAGE IIIa, step 8).