mlPEA User Manual

(version 1.0)

- mlPEA is a user-friendly and multi-functionality platform specifically tailored to the needs of streamlined processing of m^{6}A-Seq data in a reference genome-free manner. By taking advantage of machine learning (ML) algorithms, mlPEA enhanced the m^{6}A-Seq data analysis by constructing robust computational models for identifying high-quality transcripts and high-confidence m^{6}A-modified regions.
- mlPEA comprises four functional modules: **Data Preprocessing, Transcriptome Construction, m^{6}A Calling, and Functional Exploration**.
- mlPEA was powered with an advanced packaging technology, which enables compatibility and portability.
- mlPEA project is hosted on http://github.com/cma2015/mlPEA
- mlPEA docker image is available at http://hub.docker.com/r/malab/mlpea

Transcriptome Construction Module

This module provides step-by-step functions required for transcriptome construction.

Transcriptome Assembly

Four assemblers that support strand-specific m^{6}A-Seq data are wrapped to use. Currently, **Trinity**, **rnaSPAdes**, **TransABySS**, **TransLiG**.

Tools	Description	Input	Output	Time (test data)	Reference
Trinity	Trinity partitions the sequence data into many individual de Bruijn graphs, each representing the transcriptional complexity at a given gene or locus, and then processes each graph independently to extract full-length splicing isoforms and to tease apart transcripts derived from paralogous genes.	Sequencing reads in FASTQ format	Transcripts in FASTA format	~4 mins	(Grabherr *et al.*, 2011)
rnaSPAdes	rnaSPAdes has been developed on top of the SPAdes genome assembler and explores computational parallels between assembly of transcriptomes and single-cell genomes.	Sequencing reads in FASTQ format	Transcripts in FASTA format	~3 mins	(Prjibelski *et al.*, 2020)
TransABySS	A *de novo* short-read transcriptome assembly and analysis pipeline that addresses variation in local read densities by assembling read substrings with varying stringencies and then merging the resulting contigs before analysis.	Sequencing reads in FASTQ format	Transcripts in FASTA format	~5 mins	(Robertson *et al.*, 2010)
TransLiG	TransLiG is shown to be significantly superior to all the salient *de novo* assemblers in both accuracy and computing resources when tested on artificial and real RNA-seq data.	Sequencing reads in FASTQ format	Transcripts in FASTA format	~4 mins	(Liu *et al.*, 2019)

Transcriptome Screening

Transcriptome screening implements three pipelines for transcripts deduplication, respectively.

Tools	Description	Input	Output	Time (test data)	Reference
Merge and Deduplication	A novel program CD-HIT for clustering biological sequences to reduce sequence redundancy and improve the performance of other sequence analyses.	Multiple Transcripts files in FASTA format	Transcripts in FASTA format	~5 mins	(Fu *et al.*, 2012)
Reads Coverage-based Screening	An ultrafast and memory-efficient tool Bowtie 2 for aligning sequencing reads, reads coverage was measured by SAMtools and BEDTools.	Transcripts in FASTA format	Transcripts in FASTA format	~20 mins	(Langmead *et al.*, 2009; Langmead *et al.*, 2012) (Danecek *et al.*, 2021) (Quinlan *et al.*, 2010)
Machine Learning-based Screening	An ML-based classification model to distinguish high-quality transcripts from noisy ones using the random forest-based PSoL algorithm, which requires only a set of positive samples. The positive sample set is constructed by clustering analysis of assembled transcripts and already annotated mRNAs using a fast and sensitive sequence alignment method MMseqs2. It provides more than 670 features to encode each transcript sequence.	Transcripts in FASTA format	Transcripts in FASTA format	~10mins	(Steinegger *et al.*, 2017) (Wang *et al.*, 2023) (Wang *et al.*, 2006)

Transcriptome Assembly

Currently, mlPEA wrapped four assemblers that support strand-specific m^{6}A-Seq data to *de nove* assemble transcripts. Here, we take Trinity as an example to show how to use mlPEA to run transcriptome assembly, the other three assemblers are similar.

Input

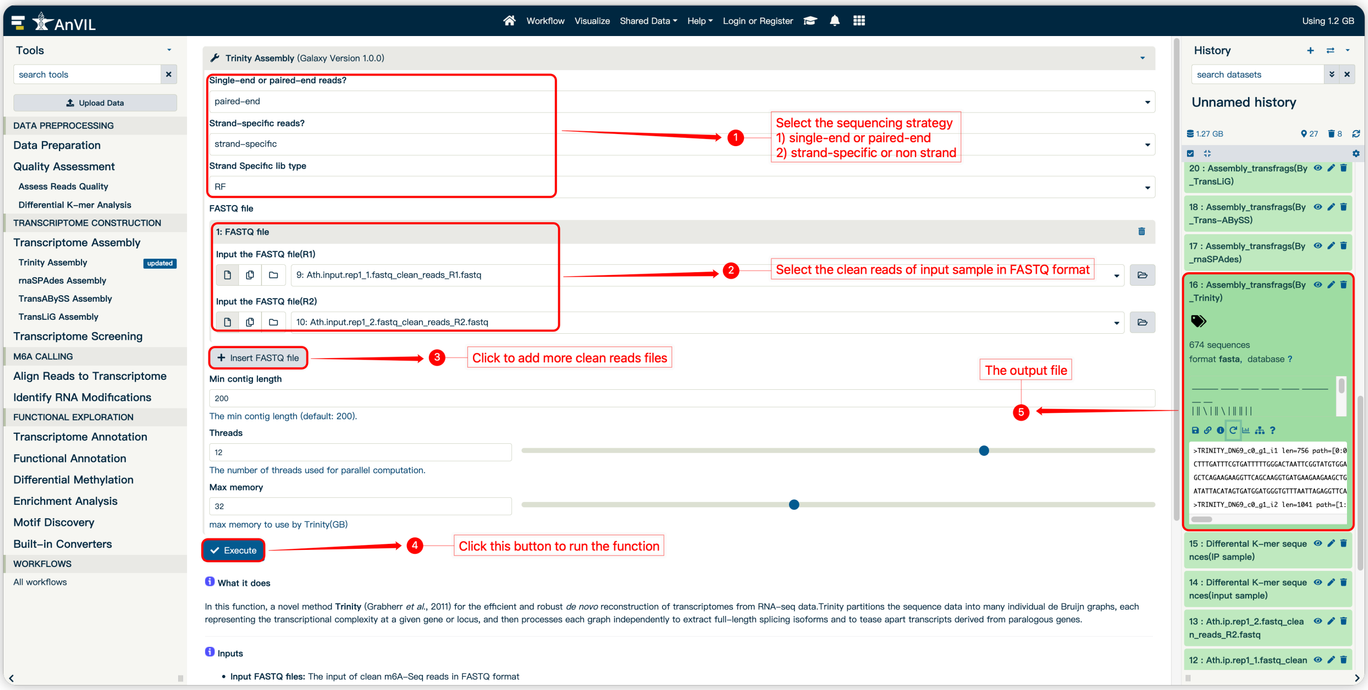
- **Input FASTQ files:** cleaned m^{6}A-Seq input reads in FASTQ format

Output

- ***De novo* assembled transcripts by Trinity in FASTA format**

How to use this function

- The following screenshot shows us how to use this function



```
>TRINITY_DN69_c0_g1_i1 len=756 path=[0:0
CTTTGATTTCGTGATTTTTGGGACTAATTCGGTATGTGGA
GCTCAGAAGAAGGTTCAGCAAGGTGATGAAGAAGAAGCTG
ATATTACATAGTGATGGATGGGTGTTTAATTAGAGGTTCA
>TRINITY_DN69_c0_g1_i2 len=1041 path=[1:
```

Trinity Assembly (Galaxy Version 1.0.0)

Single-end or paired-end reads?
paired-end

Strand-specific reads?
strand-specific

Strand Specific lib type
RF

1 Select the sequencing strategy
1) single-end or paired-end
2) strand-specific or non strand

FASTQ file

1: FASTQ file

Input the FASTQ file(R1)
9: Ath.input.rep1_1.fastq_clean_reads_R1.fastq

Input the FASTQ file(R2)
10: Ath.input.rep1_2.fastq_clean_reads_R2.fastq

2 Select the clean reads of input sample in FASTQ format

+ Insert FASTQ file

3 Click to add more clean reads files

Min contig length
200
The min contig length (default: 200).

The output file 5

Threads
12
The number of threads used for parallel computation.

Max memory
32
max memory to use by Trinity(GB)

✓ Execute

4 Click this button to run the function

What it does

In this function, a novel method **Trinity** (Grabherr *et al.*, 2011) for the efficient and robust *de novo* reconstruction of transcriptomes from RNA-seq data.Trinity partitions the sequence data into many individual de Bruijn graphs, each representing the transcriptional complexity at a given gene or locus, and then processes each graph independently to extract full-length splicing isoforms and to tease apart transcripts derived from paralogous genes.

Inputs

- **Input FASTQ files:** The input of clean m6A-Seq reads in FASTQ format

Merge and Deduplication

In this function, a novel program **CD-HIT** for clustering biological sequences to reduce sequence redundancy and improve the performance of other sequence analyses.

Input

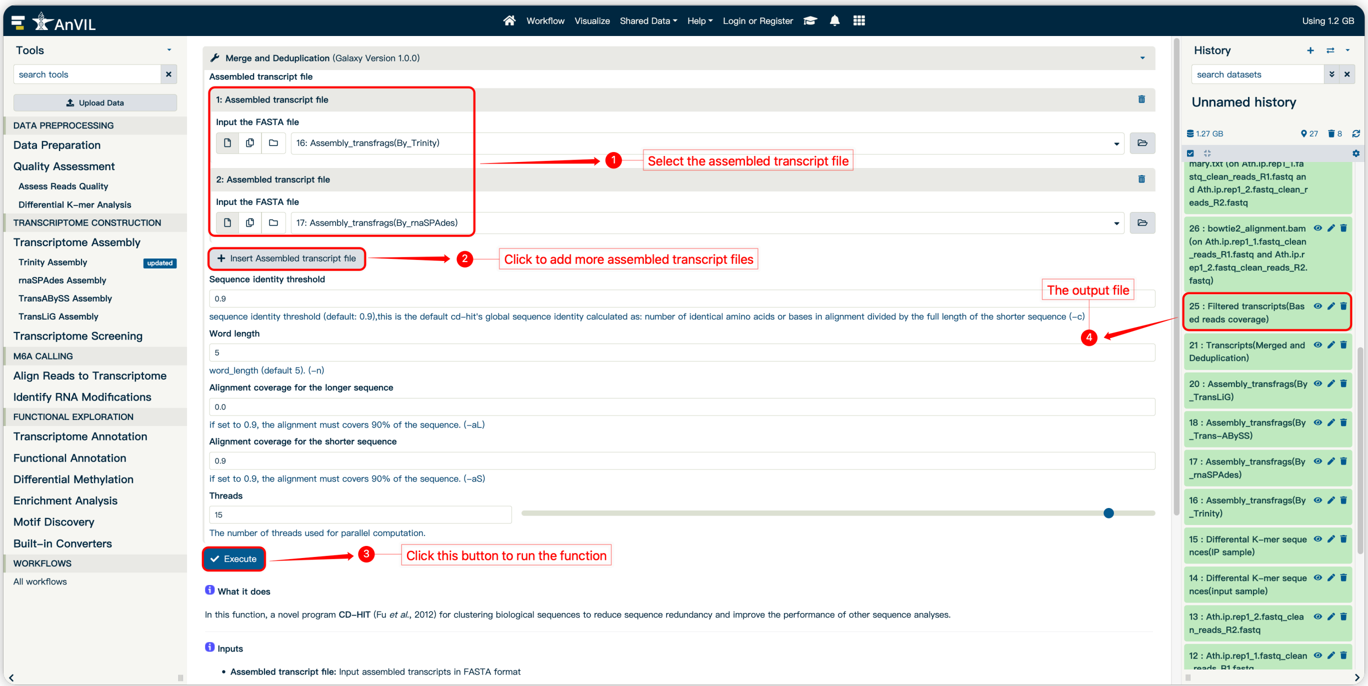
- **Input FASTA files:** assembled transcripts in FASTA format

Output

- **Transcripts after deduplication in FASTA format**

How to use this function

- The following screenshot shows us how to use this function


AnVIL
Workflow
Visualize
Shared Data
Help
Login or Register
Using 1.2 GB
Tools
search tools
Upload Data
DATA PREPROCESSING
Data Preparation
Quality Assessment
Assess Reads Quality
Differential K–mer Analysis
TRANSCRIPTOME CONSTRUCTION
Transcriptome Assembly
Trinity Assembly
updated
rnaSPAdes Assembly
TransABySS Assembly
TransLiG Assembly
Transcriptome Screening
M6A CALLING
Align Reads to Transcriptome
Identify RNA Modifications
FUNCTIONAL EXPLORATION
Transcriptome Annotation
Functional Annotation
Differential Methylation
Enrichment Analysis
Motif Discovery
Built–in Converters
WORKFLOWS
All workflows
History
search datasets
Unnamed history
1.27 GB
27
8
mary.txt (on Ath.ip.rep1_1.fastq_clean_reads_R1.fastq and Ath.ip.rep1_2.fastq_clean_reads_R2.fastq
26 : bowtie2_alignment.bam (on Ath.ip.rep1_1.fastq_clean_reads_R1.fastq and Ath.ip.rep1_2.fastq_clean_reads_R2.fastq)
25 : Filtered transcripts(Based reads coverage)
21 : Transcripts(Merged and Deduplication)
20 : Assembly_transfrags(By_TransLiG)
18 : Assembly_transfrags(By_Trans–ABySS)
17 : Assembly_transfrags(By_rnaSPAdes)
16 : Assembly_transfrags(By_Trinity)
15 : Differental K–mer sequences(IP sample)
14 : Differental K–mer sequences(input sample)
13 : Ath.ip.rep1_2.fastq_clean_reads_R2.fastq
12 : Ath.ip.rep1_1.fastq_clean_reads_R1.fastq

Merge and Deduplication (Galaxy Version 1.0.0)

Assembled transcript file

1: Assembled transcript file

Input the FASTA file

16: Assembly_transfrags(By_Trinity)

2: Assembled transcript file

Input the FASTA file

17: Assembly_transfrags(By_rnaSPAdes)

+ Insert Assembled transcript file

Sequence identity threshold

0.9

sequence identity threshold (default: 0.9),this is the default cd–hit's global sequence identity calculated as: number of identical amino acids or bases in alignment divided by the full length of the shorter sequence (–c)

Word length

5

word_length (default 5). (–n)

Alignment coverage for the longer sequence

0.0

if set to 0.9, the alignment must covers 90% of the sequence. (–aL)

Alignment coverage for the shorter sequence

0.9

if set to 0.9, the alignment must covers 90% of the sequence. (–aS)

Threads

15

The number of threads used for parallel computation.

✓ Execute

1 Select the assembled transcript file

2 Click to add more assembled transcript files

3 Click this button to run the function

4 The output file

What it does

In this function, a novel program **CD–HIT** (Fu *et al.*, 2012) for clustering biological sequences to reduce sequence redundancy and improve the performance of other sequence analyses.

Inputs

- **Assembled transcript file:** Input assembled transcripts in FASTA format

Reads Coverage-based Screening

In this function, we developed a pipline for screening transcripts based on reads coverage. **Bowtie2** for aligning sequencing reads, reads coverage was measured by **SAMtools** and **BEDTools**.

Input

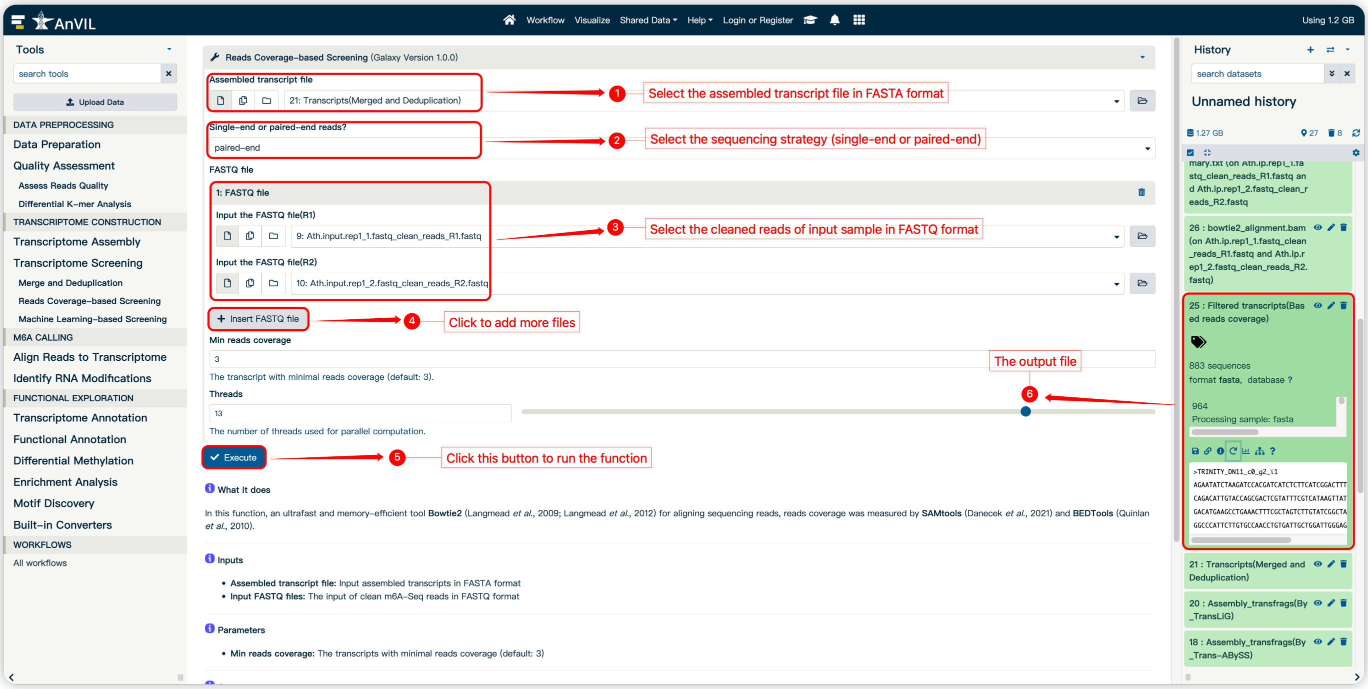
- **Input FASTQ files:** cleaned m^{6}A-Seq input reads in FASTQ format

Output

- ***De novo* assembled transcripts in FASTA format**

How to use this function

- The following screenshot shows us how to use this function



Reads Coverage–based Screening (Galaxy Version 1.0.0)

Assembled transcript file: 21: Transcripts(Merged and Deduplication)

1 Select the assembled transcript file in FASTA format

Single–end or paired–end reads?: paired–end

2 Select the sequencing strategy (single–end or paired–end)

FASTQ file

1: FASTQ file

Input the FASTQ file(R1): 9: Ath.input.rep1_1.fastq_clean_reads_R1.fastq

3 Select the cleaned reads of input sample in FASTQ format

Input the FASTQ file(R2): 10: Ath.input.rep1_2.fastq_clean_reads_R2.fastq

+ Insert FASTQ file

4 Click to add more files

Min reads coverage: 3

The transcript with minimal reads coverage (default: 3).

Threads: 13

The number of threads used for parallel computation.

6 The output file

✓ Execute

5 Click this button to run the function

What it does

In this function, an ultrafast and memory–efficient tool **Bowtie2** (Langmead *et al.*, 2009; Langmead *et al.*, 2012) for aligning sequencing reads, reads coverage was measured by **SAMtools** (Danecek *et al.*, 2021) and **BEDTools** (Quinlan *et al.*, 2010).

Inputs

- **Assembled transcript file:** Input assembled transcripts in FASTA format
- **Input FASTQ files:** The input of clean m6A–Seq reads in FASTQ format

Parameters

- **Min reads coverage:** The transcripts with minimal reads coverage (default: 3)

Machine Learning-based Screening

In this function, an ML-based classification model to distinguish high-quality transcripts from noisy ones using the random forest-based Positive Sample only Learning (PSoL) algorithm, which requires only a set of positive samples. The positive sample set is constructed by clustering analysis of assembled transcripts and already annotated mRNAs using a fast and sensitive sequence alignment method **MMseqs2**. It provides more than 670 features to encode each transcript sequence, including 177 sequence-intrinsic features, 399 physico-chemical features and 101 structure-based features.

Input

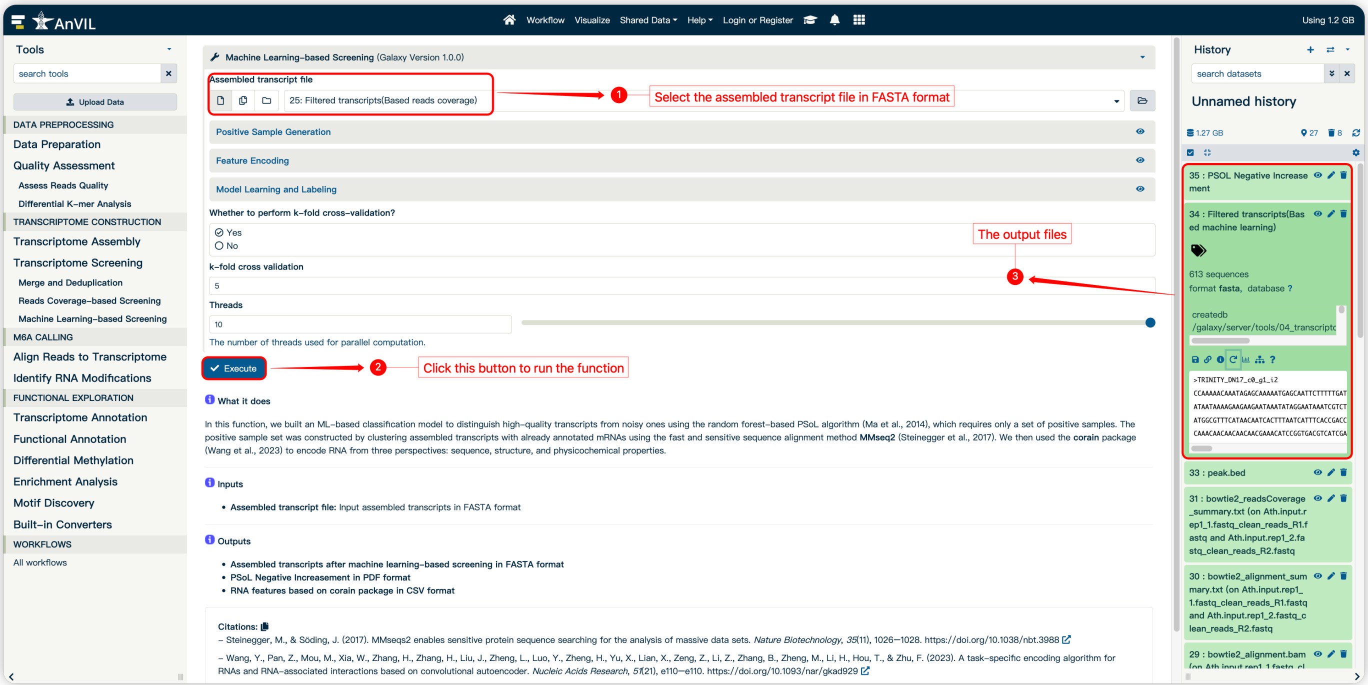
- **Input FASTQ files:** *de novo* assembled transcripts in FASTA format

Output

- **High-quality transcripts in FASTA format**
- **PSoL Negative Increasement in PDF format**
- **RNA features based on corain package in CSV format**

How to use this function

- The following screenshot shows us how to use this function



Machine Learning–based Screening (Galaxy Version 1.0.0)

Assembled transcript file

25: Filtered transcripts(Based reads coverage)

1 Select the assembled transcript file in FASTA format

Positive Sample Generation

Feature Encoding

Model Learning and Labeling

Whether to perform k–fold cross–validation?

☑ Yes
○ No

k–fold cross validation

5

Threads

10

The number of threads used for parallel computation.

2 Click this button to run the function

3 The output files

What it does

In this function, we built an ML–based classification model to distinguish high–quality transcripts from noisy ones using the random forest–based PSoL algorithm (Ma et al., 2014), which requires only a set of positive samples. The positive sample set was constructed by clustering assembled transcripts with already annotated mRNAs using the fast and sensitive sequence alignment method **MMseq2** (Steinegger et al., 2017). We then used the **corain** package (Wang et al., 2023) to encode RNA from three perspectives: sequence, structure, and physicochemical properties.

Inputs

- **Assembled transcript file:** Input assembled transcripts in FASTA format

Outputs

- **Assembled transcripts after machine learning–based screening in FASTA format**
- **PSoL Negative Increasement in PDF format**
- **RNA features based on corain package in CSV format**

Citations:

– Steinegger, M., & Söding, J. (2017). MMseqs2 enables sensitive protein sequence searching for the analysis of massive data sets. *Nature Biotechnology*, *35*(11), 1026—1028. https://doi.org/10.1038/nbt.3988

– Wang, Y., Pan, Z., Mou, M., Xia, W., Zhang, H., Zhang, H., Liu, J., Zheng, L., Luo, Y., Zheng, H., Yu, X., Lian, X., Zeng, Z., Li, Z., Zhang, B., Zheng, M., Li, H., Hou, T., & Zhu, F. (2023). A task–specific encoding algorithm for RNAs and RNA–associated interactions based on convolutional autoencoder. *Nucleic Acids Research*, *51*(21), e110—e110. https://doi.org/10.1093/nar/gkad929

35 : PSOL Negative Increase ment

34 : Filtered transcripts(Bas ed machine learning)

613 sequences
format **fasta**, database ?

```
createdb
/galaxy/server/tools/04_transcripto
```

```
>TRINITY_DN17_c0_g1_i2
CCAAAAACAAATAGAGCAAAAATGAGCAATTCTTTTTGAT
ATAATAAAAGAAGAAGAATAAATATAGGAATAAATCGTCT
ATGGCGTTTCATAACAATCACTTTAATCATTTCACCGACC
CAAACAACAACAACAACGAAACATCCGGTGACGTCATCGA
```

33 : peak.bed

31 : bowtie2_readsCoverage_summary.txt (on Ath.input.rep1_1.fastq_clean_reads_R1.fastq and Ath.input.rep1_2.fastq_clean_reads_R2.fastq

30 : bowtie2_alignment_summary.txt (on Ath.input.rep1_1.fastq_clean_reads_R1.fastq and Ath.input.rep1_2.fastq_clean_reads_R2.fastq

29 : bowtie2_alignment.bam (on Ath.input.rep1_1.fastq_cl