PEA-m6A User Manual (version 1.0)

- PEA-m6A is an ensemble learning framework for predicting m6A modifications at regional-scale.
- PEA-m6A consists of four modules:**Sample Preparation, Feature Encoding, Model Development and Model Assessment**, each of which contains a comprehensive collection of functions with pre-specified parameters available.
- PEA-m6A was powered with an advanced packaging technology, which enables compatibility and portability.
- PEA-m6A project is hosted on http://github.com/cma2015/PEA-m6A
- PEA-m6A docker image is available at http://hub.docker.com/r/malab/peam6a
- PEA-m6A server can be accessed via http://peam6a.omstudio.cloud

Sample Preparation

This module provides seven functions (see following table for details) to prepare epittranscriptome training data.

Tools	Description	Input	Output	Reference
Sequence Data Preprocessing	Convert epitranscriptome sequencing reads from SRA to FASTQ format	Epitranscriptome sequencing reads in SRA format	Epitranscriptome sequencing reads in FASTQ format	SRA Toolkit
Assess Reads Quality	This function firstly performs quality control using FastQC and then trims low-quality reads using fastp	Epitranscriptome sequencing reads in FASTQ format	Clean reads in FASTQ format; Reads quality report in HTML format	FastQC,fastp
HISAT2	HISAT2 is an ultrafast spliced aligner with low memory requirements. It supports genomes of any size, including those larger than 4 billion bases	Epitranscriptome sequencing reads in FASTQ format and reference genome sequences in FASTA format	Read alignments in SAM/BAM format	Kim *et al*., 2015, Nature Methods
Peak Calling from the MeRIP-Seq data	Identify enriched genomic regions from MeRIP-Seq experiment	Read alignments of IP and input in SAM/BAM format and reference genome sequences in FASTA format	RNA modifications in BED format	PEA
RNA Modifications Annotation with Gene	Link RNA modifications to nearest genes based on genomic coordinate	RNA modifications in BED format and genome annotation in GTF/GFF3 format	Detailed RNA modifications-related genes	In-house scripts
Motif Analysis	Integrate MEME-ChIP and DREME to perform de-novo motif discovery	RNA modifications in BED format and reference genome sequences in FASTA format	Discovered motifs in HTML format	Timothy *et al*., 2011, Bioinformatics,Philip *et al*., 2011, Bioinformatics,Heinz *et al*., 2010, Molecular Cell
Sample Generation	Extraction postive and negative sequences for training in FASTA format	RNA modifications in BED format, reference genome sequences in FASTA format, gene and exons annotation in BED format	Postive and neagtive samples in BED format and in FASTA format	In-house scripts

Sequence Data Preprocessing

This function wrapped **fastq-dump** function implemented in SRA Toolkit. See http://trace.ncbi.nlm.nih.gov/Traces/sra/sra.cgi?view=software for details.

Input

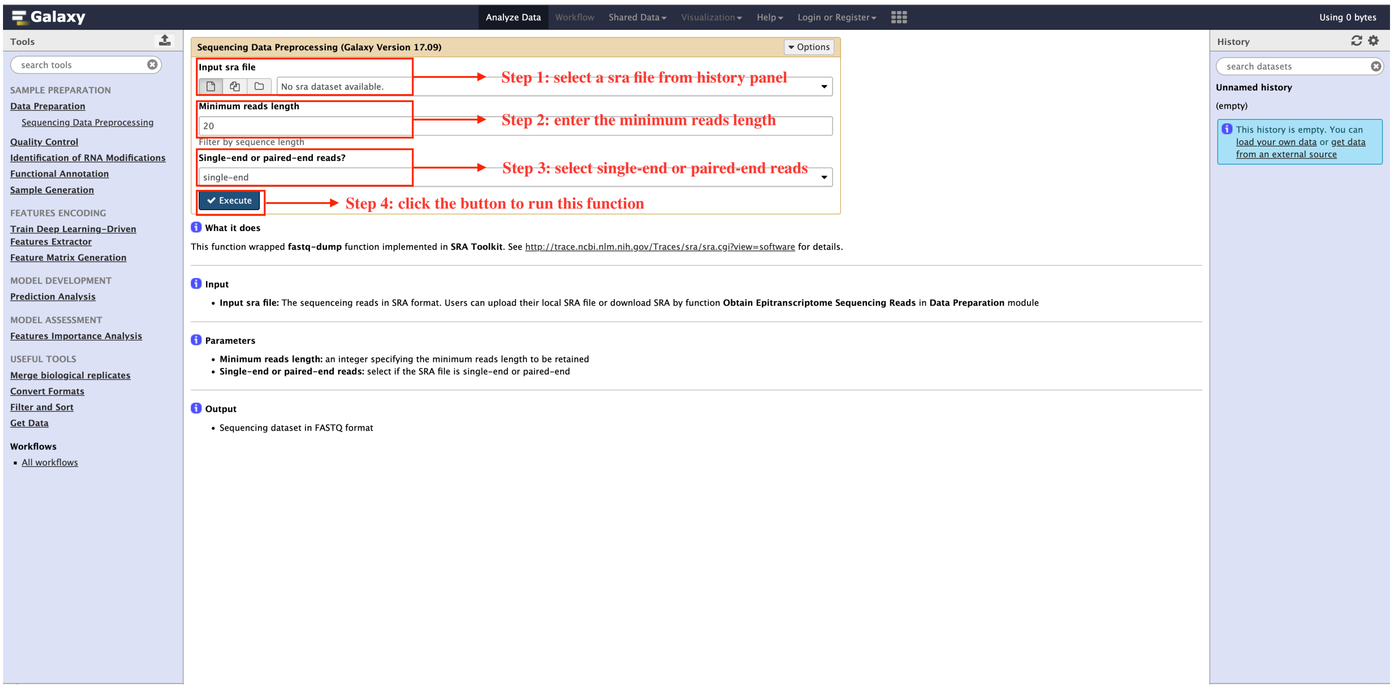
- **Input sra file:** The sequenceing reads in SRA format. Users can upload their local SRA file or download SRA by function **Obtain Epitranscriptome Sequencing Reads** in **Data Preparation** module

Output

- Sequencing dataset in FASTQ format

How to use this function

- The following screenshot shows us how to use this function to convert sequencing reads in SRA format to FASTQ format



Sequencing Data Preprocessing (Galaxy Version 17.09)

Input sra file: No sra dataset available.

Step 1: select a sra file from history panel

Minimum reads length: 20

Filter by sequence length

Step 2: enter the minimum reads length

Single-end or paired-end reads?: single-end

Step 3: select single-end or paired-end reads

Step 4: click the button to run this function

What it does

This function wrapped **fastq-dump** function implemented in **SRA Toolkit**. See http://trace.ncbi.nlm.nih.gov/Traces/sra/sra.cgi?view=software for details.

Input

- **Input sra file:** The sequenceing reads in SRA format. Users can upload their local SRA file or download SRA by function **Obtain Epitranscriptome Sequencing Reads** in **Data Preparation** module

Parameters

- **Minimum reads length:** an integer specifying the minimum reads length to be retained
- **Single-end or paired-end reads:** select if the SRA file is single-end or paired-end

Output

- Sequencing dataset in FASTQ format

This history is empty. You can load your own data or get data from an external source

Assess Reads Quality

In this function, two existing NGS tools **FastQC** (Andrews *et al*., 2010) and **fastp** (Chen *et al*., 2018) are integrated to check sequencing reads quality and obtain high-quality reads, respectively.

Input

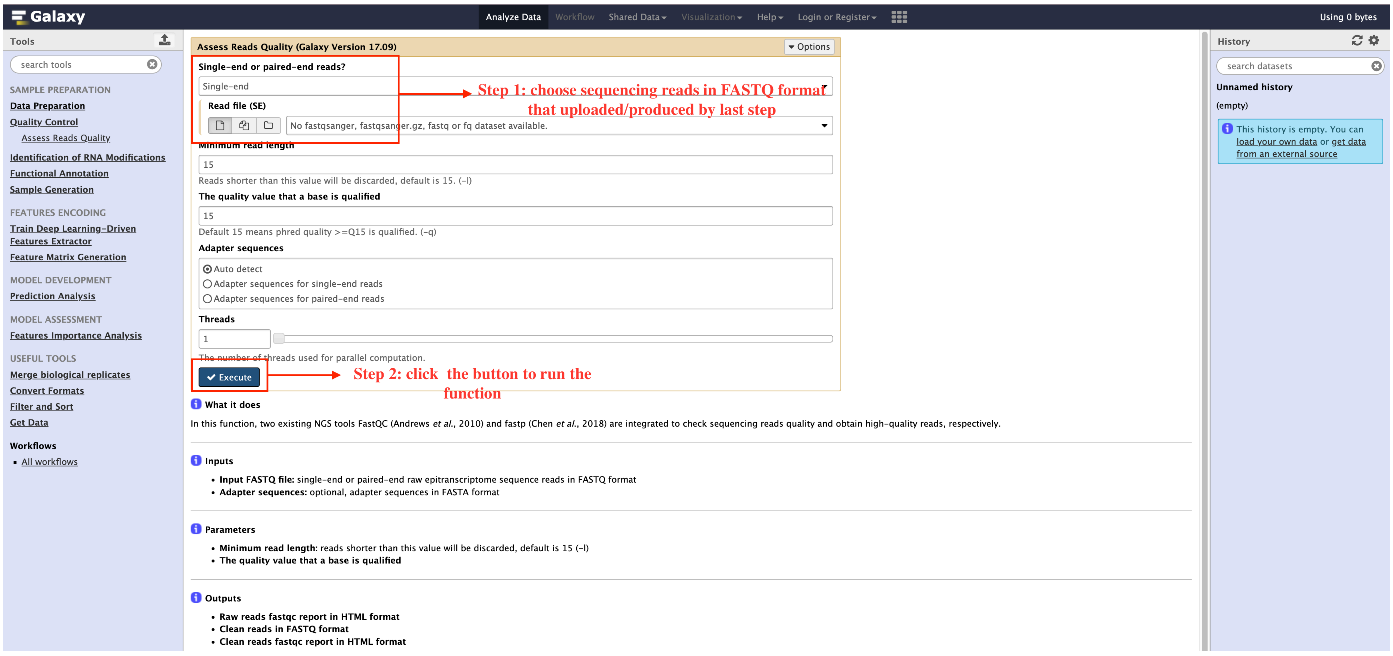
- **Input FASTQ file**: single-end or paired-end raw epitranscriptome sequence reads in FASTQ format
- **Adapter sequences**: optional, adapter sequences in FASTA format

Output

- **Clean reads in FASTQ format**
- **Reads quality report in HTML format**

How to use this function

- The following screenshot shows us how to assess reads quality



Assess Reads Quality (Galaxy Version 17.09)

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

Read file (SE)
No fastqsanger, fastqsanger.gz, fastq or fq dataset available.

Step 1: choose sequencing reads in FASTQ format that uploaded/produced by last step

Minimum read length
15
Reads shorter than this value will be discarded, default is 15. (-l)

The quality value that a base is qualified
15
Default 15 means phred quality >=Q15 is qualified. (-q)

Adapter sequences
● Auto detect
○ Adapter sequences for single-end reads
○ Adapter sequences for paired-end reads

Threads
1
The number of threads used for parallel computation.

Step 2: click the button to run the function

What it does

In this function, two existing NGS tools FastQC (Andrews *et al.*, 2010) and fastp (Chen *et al.*, 2018) are integrated to check sequencing reads quality and obtain high-quality reads, respectively.

Inputs

- **Input FASTQ file:** single-end or paired-end raw epitranscriptome sequence reads in FASTQ format
- **Adapter sequences:** optional, adapter sequences in FASTA format

Parameters

- **Minimum read length:** reads shorter than this value will be discarded, default is 15 (-l)
- **The quality value that a base is qualified**

Outputs

- **Raw reads fastqc report in HTML format**
- **Clean reads in FASTQ format**
- **Clean reads fastqc report in HTML format**

Unnamed history
(empty)
This history is empty. You can load your own data or get data from an external source

Align reads to genome

In this function, PEA-m6A adopted HISAT2 as the aligners to map epitranscriptome reads to genome.

Input

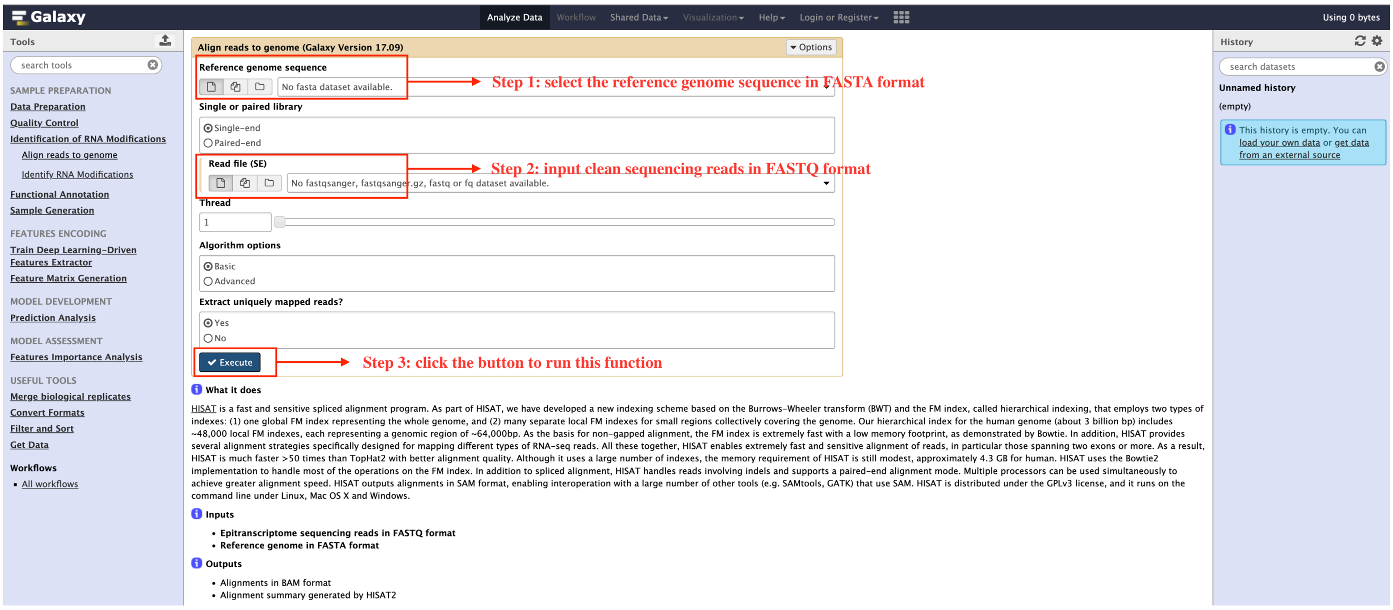
- **Epitranscriptome sequencing reads in FASTQ format**
- **Reference genome in FASTA format**

Output

- Alignments in BAM format
- Alignment summary generated by HISAT2

How to use this function

- The following screenshot to run this function



Align reads to genome (Galaxy Version 17.09)

Reference genome sequence: No fasta dataset available.

Step 1: select the reference genome sequence in FASTA format

Single or paired library: ● Single-end ○ Paired-end

Read file (SE): No fastqsanger, fastqsanger.gz, fastq or fq dataset available.

Step 2: input clean sequencing reads in FASTQ format

Thread: 1

Algorithm options: ● Basic ○ Advanced

Extract uniquely mapped reads? ● Yes ○ No

Step 3: click the button to run this function

What it does

HISAT is a fast and sensitive spliced alignment program. As part of HISAT, we have developed a new indexing scheme based on the Burrows-Wheeler transform (BWT) and the FM index, called hierarchical indexing, that employs two types of indexes: (1) one global FM index representing the whole genome, and (2) many separate local FM indexes for small regions collectively covering the genome. Our hierarchical index for the human genome (about 3 billion bp) includes ~48,000 local FM indexes, each representing a genomic region of ~64,000bp. As the basis for non-gapped alignment, the FM index is extremely fast with a low memory footprint, as demonstrated by Bowtie. In addition, HISAT provides several alignment strategies specifically designed for mapping different types of RNA-seq reads. All these together, HISAT enables extremely fast and sensitive alignment of reads, in particular those spanning two exons or more. As a result, HISAT is much faster >50 times than TopHat2 with better alignment quality. Although it uses a large number of indexes, the memory requirement of HISAT is still modest, approximately 4.3 GB for human. HISAT uses the Bowtie2 implementation to handle most of the operations on the FM index. In addition to spliced alignment, HISAT handles reads involving indels and supports a paired-end alignment mode. Multiple processors can be used simultaneously to achieve greater alignment speed. HISAT outputs alignments in SAM format, enabling interoperation with a large number of other tools (e.g. SAMtools, GATK) that use SAM. HISAT is distributed under the GPLv3 license, and it runs on the command line under Linux, Mac OS X and Windows.

Inputs

- **Epitranscriptome sequencing reads in FASTQ format**
- **Reference genome in FASTA format**

Outputs

- Alignments in BAM format
- Alignment summary generated by HISAT2

Unnamed history

(empty)

This history is empty. You can load your own data or get data from an external source

Peak calling from the MeRIP-Seq data

Peak calling is used to identify enriched genomic regions in MeRIP-seq or ChIP-seq experiments. The function is implemented using the **peakCalling** function in PEA package (zhai *et al*., 2018)

Input

- **IP sample:** The IP experiment in BAM format
- **Input sample:** The input control experiment in BAM format
- **Reference genome:** The Reference genome sequences with FASTA format
- **Reference annotation file:** The Reference genome annotation file with GTF/GFF3 format (required for methods: **exomePeak**)

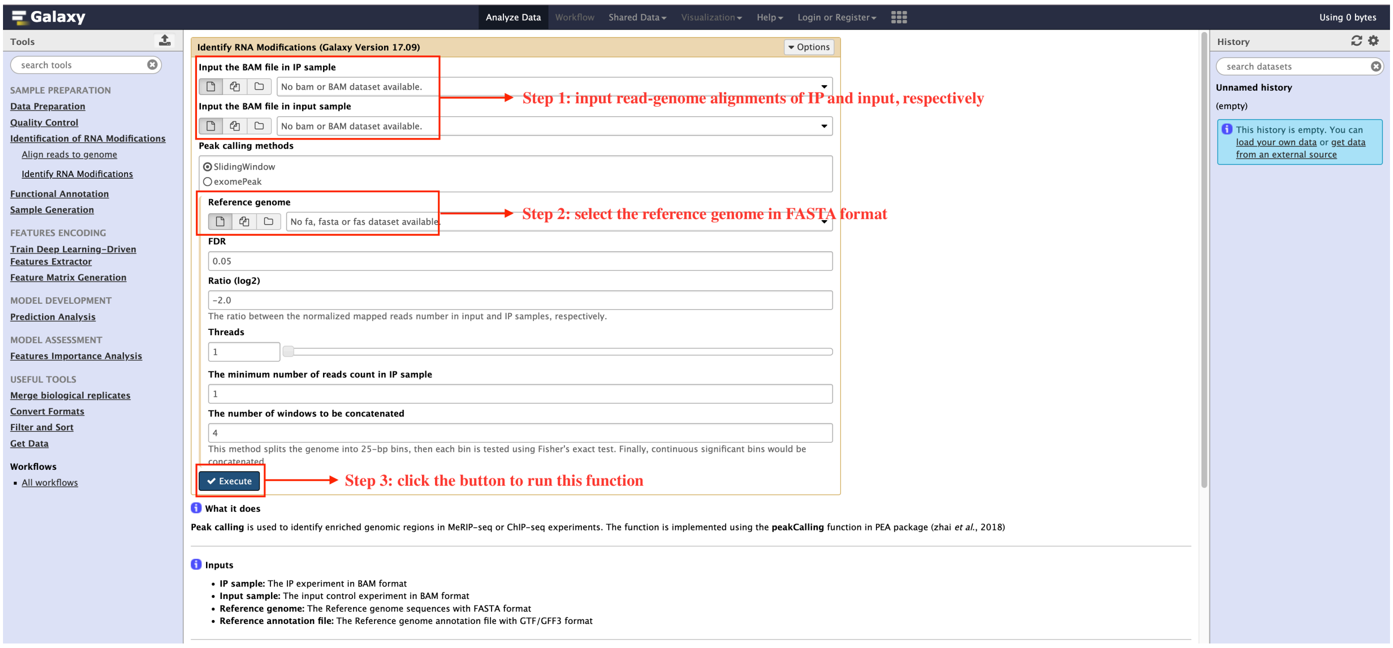
Output

- **The enriched peak region matrix in BED format**

Chromosome	Start(1-based)	End	Bin number	Mean FDR	Max FDR	Minimum FDR	Mean Ratio	Max Ratio	Minimum Ratio
1	67476	67575	4	0.0136	0.0328	0.0001	-1.0012	-0.6334	-1.581
1	330776	330875	4	0.0215	0.0381	0.0007	-1.576	-1.4077	-1.788
1	389201	389300	4	0.0024	0.0070	0.0002	-1.115	-1.0598	-1.190

How to use this function

- The following screenshot to run this function.



Identify RNA Modifications (Galaxy Version 17.09)

Input the BAM file in IP sample: No bam or BAM dataset available.

Input the BAM file in input sample: No bam or BAM dataset available.

Step 1: input read-genome alignments of IP and input, respectively

Peak calling methods: ● SlidingWindow ○ exomePeak

Reference genome: No fa, fasta or fas dataset available.

Step 2: select the reference genome in FASTA format

FDR: 0.05

Ratio (log2): -2.0

The ratio between the normalized mapped reads number in input and IP samples, respectively.

Threads: 1

The minimum number of reads count in IP sample: 1

The number of windows to be concatenated: 4

This method splits the genome into 25-bp bins, then each bin is tested using Fisher's exact test. Finally, continuous significant bins would be concatenated

Step 3: click the button to run this function

What it does

Peak calling is used to identify enriched genomic regions in MeRIP-seq or ChIP-seq experiments. The function is implemented using the **peakCalling** function in PEA package (zhai *et al.*, 2018)

Inputs

- **IP sample:** The IP experiment in BAM format
- **Input sample:** The input control experiment in BAM format
- **Reference genome:** The Reference genome sequences with FASTA format
- **Reference annotation file:** The Reference genome annotation file with GTF/GFF3 format

RNA Modifications Annotation with Gene

This function is designed to annotate RNA modifications with genes, users can specify the minimum overlapped length with genes.

Input

- **RNA modifications:** RNA modifications in BED format which can be obtained by any function in **Identify RNA Modifications**
- **Genome annotation in GTF/GFF3 format:** The genome annotation in GTF/GFF3 format

Output

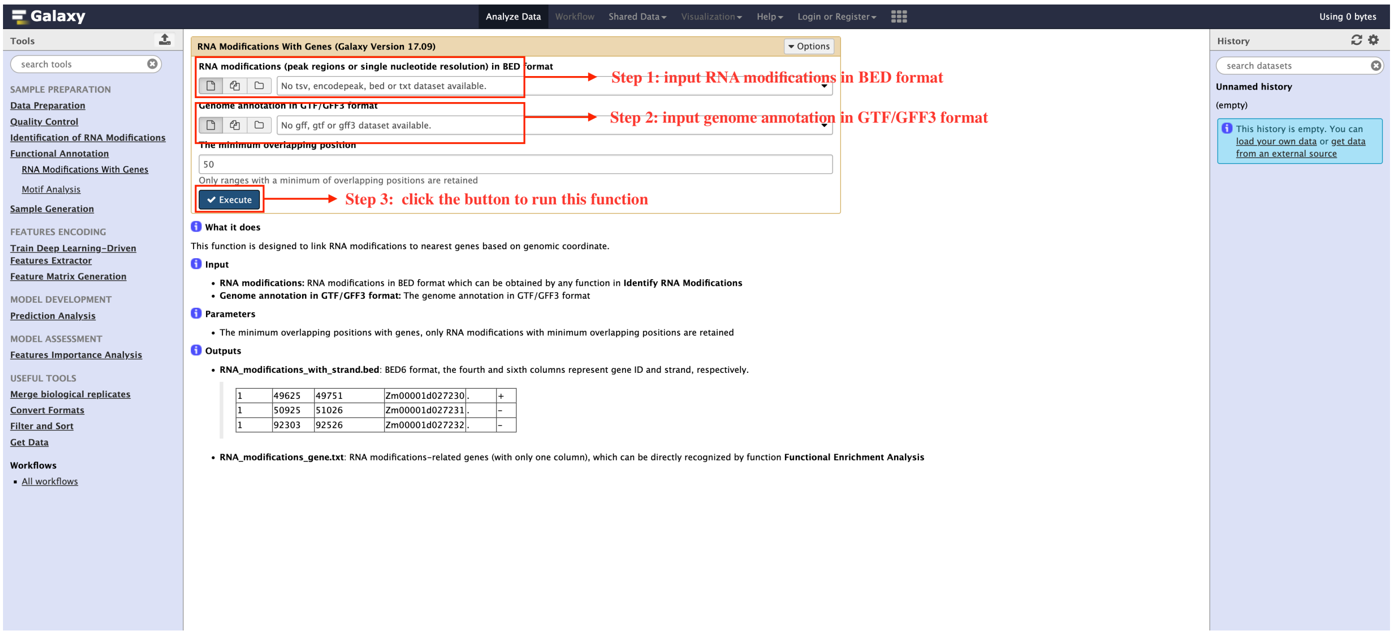
- **RNA_modifications_with_strand.bed**: BED6 format, the fourth and sixth columns represent gene ID and strand, respectively.

Chr	Start	End	GeneID	Un	Strand
1	49625	49751	Zm00001d027230	.	+
1	50925	51026	Zm00001d027231	.	-
1	92303	92526	Zm00001d027232	.	-

- **RNA_modifications_gene.txt**: RNA modifications-related genes (with only one column)

How to use this function

- The following screenshot to run this function.



RNA Modifications With Genes (Galaxy Version 17.09)

RNA modifications (peak regions or single nucleotide resolution) in BED format

No tsv, encodepeak, bed or txt dataset available.

Step 1: input RNA modifications in BED format

Genome annotation in GTF/GFF3 format

No gff, gtf or gff3 dataset available.

Step 2: input genome annotation in GTF/GFF3 format

The minimum overlapping position

50

Only ranges with a minimum of overlapping positions are retained

Step 3: click the button to run this function

What it does

This function is designed to link RNA modifications to nearest genes based on genomic coordinate.

Input

- **RNA modifications:** RNA modifications in BED format which can be obtained by any function in **Identify RNA Modifications**
- **Genome annotation in GTF/GFF3 format:** The genome annotation in GTF/GFF3 format

Parameters

- The minimum overlapping positions with genes, only RNA modifications with minimum overlapping positions are retained

Outputs

- **RNA_modifications_with_strand.bed**: BED6 format, the fourth and sixth columns represent gene ID and strand, respectively.

1	49625	49751	Zm00001d027230	.	+
1	50925	51026	Zm00001d027231	.	-
1	92303	92526	Zm00001d027232	.	-

- **RNA_modifications_gene.txt**: RNA modifications-related genes (with only one column), which can be directly recognized by function **Functional Enrichment Analysis**

This history is empty. You can load your own data or get data from an external source

Motif Analysis

This function integrates MEME-ChIP and DREME to perform *de-novo* motif discovery.

Input

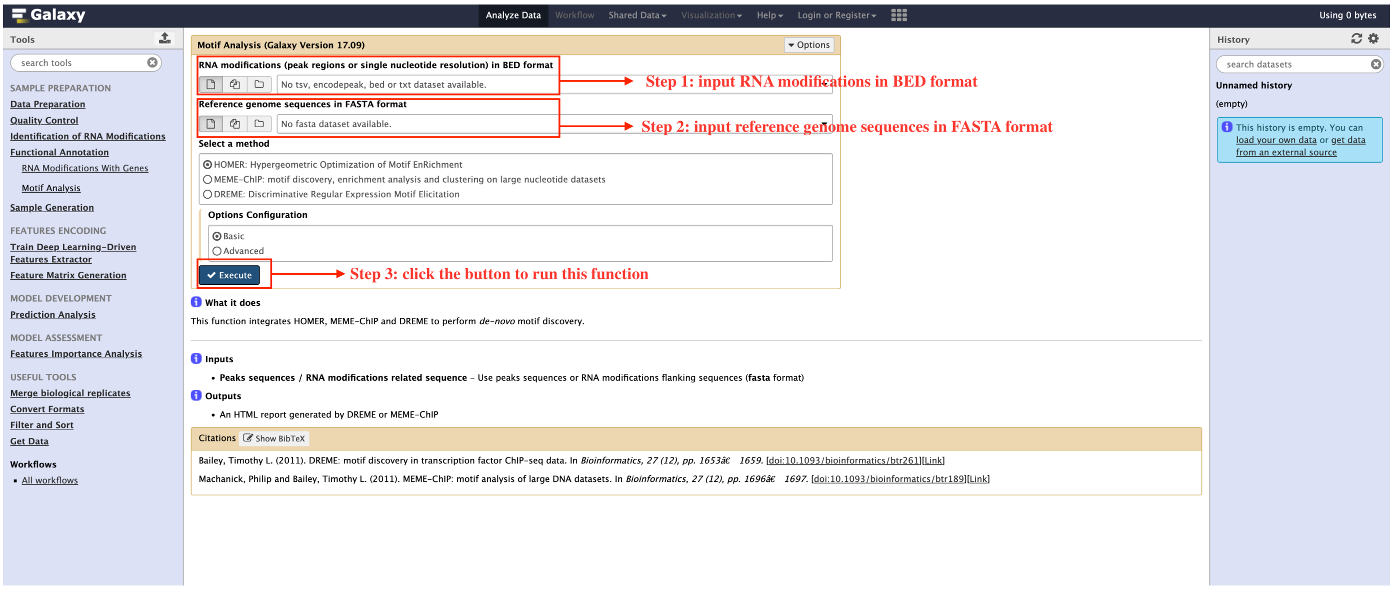
- **RNA modifications (peak regions or single nucleotide resolution) in BED format**
- **Reference genome sequences in FASTA format**

Output

- An HTML report generated by DREME or MEME-ChIP

How to use this function

- The following screenshot to run this function



Motif Analysis (Galaxy Version 17.09)

RNA modifications (peak regions or single nucleotide resolution) in BED format

No tsv, encodepeak, bed or txt dataset available.

Step 1: input RNA modifications in BED format

Reference genome sequences in FASTA format

No fasta dataset available.

Step 2: input reference genome sequences in FASTA format

Select a method

● HOMER: Hypergeometric Optimization of Motif EnRichment
○ MEME-ChIP: motif discovery, enrichment analysis and clustering on large nucleotide datasets
○ DREME: Discriminative Regular Expression Motif Elicitation

Options Configuration

● Basic
○ Advanced

Step 3: click the button to run this function

What it does

This function integrates HOMER, MEME-ChIP and DREME to perform *de-novo* motif discovery.

Inputs

- **Peaks sequences / RNA modifications related sequence** - Use peaks sequences or RNA modifications flanking sequences (**fasta** format)

Outputs

- An HTML report generated by DREME or MEME-ChIP

Citations

Bailey, Timothy L. (2011). DREME: motif discovery in transcription factor ChIP-seq data. In *Bioinformatics, 27 (12), pp. 1653â€   1659.* [doi:10.1093/bioinformatics/btr261][Link]

Machanick, Philip and Bailey, Timothy L. (2011). MEME-ChIP: motif analysis of large DNA datasets. In *Bioinformatics, 27 (12), pp. 1696â€   1697.* [doi:10.1093/bioinformatics/btr189][Link]

This history is empty. You can load your own data or get data from an external source

Sample Generation

This function was designed to generate positive and negative samples based RNA modification regions. To be specific, this function takes RNA modification regions in BED format, genomic sequences in FASTA format, gene and exon annotaiton in BED format as input, then searches consensus motif (e.g. RRACH) in the RNA modification regions and treat them as positive samples, the remaining consensus motif in the same transcript of positive samples are randomly selected as negative samples.

Input

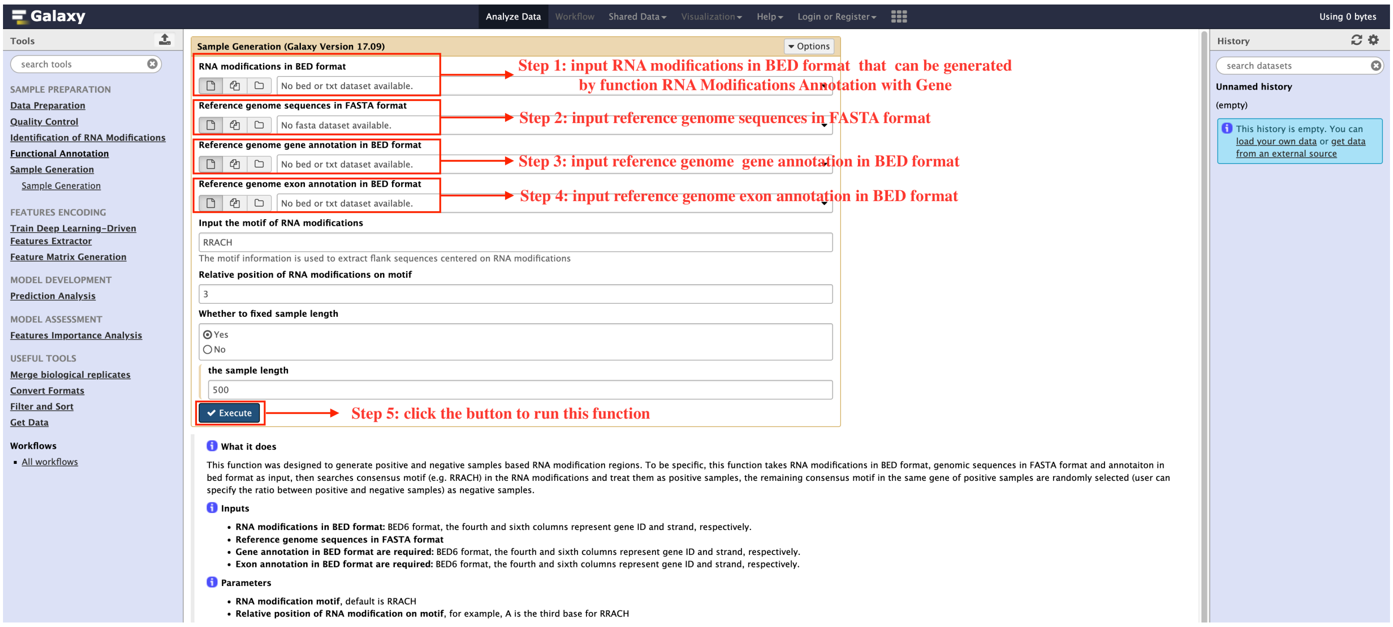
- **RNA modifications in BED format**
- **Reference genome sequences in FASTA format**
- **Genome annotation in GTF/GFF3 format are required**

Output

- **positive_samples.bed**: positive samples in BED format with 6 columns
- **positive_samples.fasta**: positive samples in FASTA format
- **negative_samples.bed**: negative samples in BED format with 6 columns
- **neagtive_samples.fasta**: negative samples in FASTA format

How to use this function

- The following screenshot to run this function



Sample Generation (Galaxy Version 17.09)

RNA modifications in BED format: No bed or txt dataset available.

Step 1: input RNA modifications in BED format that can be generated by function RNA Modifications Annotation with Gene

Reference genome sequences in FASTA format: No fasta dataset available.

Step 2: input reference genome sequences in FASTA format

Reference genome gene annotation in BED format: No bed or txt dataset available.

Step 3: input reference genome gene annotation in BED format

Reference genome exon annotation in BED format: No bed or txt dataset available.

Step 4: input reference genome exon annotation in BED format

Input the motif of RNA modifications: RRACH

The motif information is used to extract flank sequences centered on RNA modifications

Relative position of RNA modifications on motif: 3

Whether to fixed sample length

● Yes
○ No

the sample length: 500

Step 5: click the button to run this function

What it does

This function was designed to generate positive and negative samples based RNA modification regions. To be specific, this function takes RNA modifications in BED format, genomic sequences in FASTA format and annotaiton in bed format as input, then searches consensus motif (e.g. RRACH) in the RNA modifications and treat them as positive samples, the remaining consensus motif in the same gene of positive samples are randomly selected (user can specify the ratio between positive and negative samples) as negative samples.

Inputs

- **RNA modifications in BED format:** BED6 format, the fourth and sixth columns represent gene ID and strand, respectively.
- **Reference genome sequences in FASTA format**
- **Gene annotation in BED format are required:** BED6 format, the fourth and sixth columns represent gene ID and strand, respectively.
- **Exon annotation in BED format are required:** BED6 format, the fourth and sixth columns represent gene ID and strand, respectively.

Parameters

- **RNA modification motif**, default is RRACH
- **Relative position of RNA modification on motif**, for example, A is the third base for RRACH