SATAY quick guide

Contents

Purpose of the software

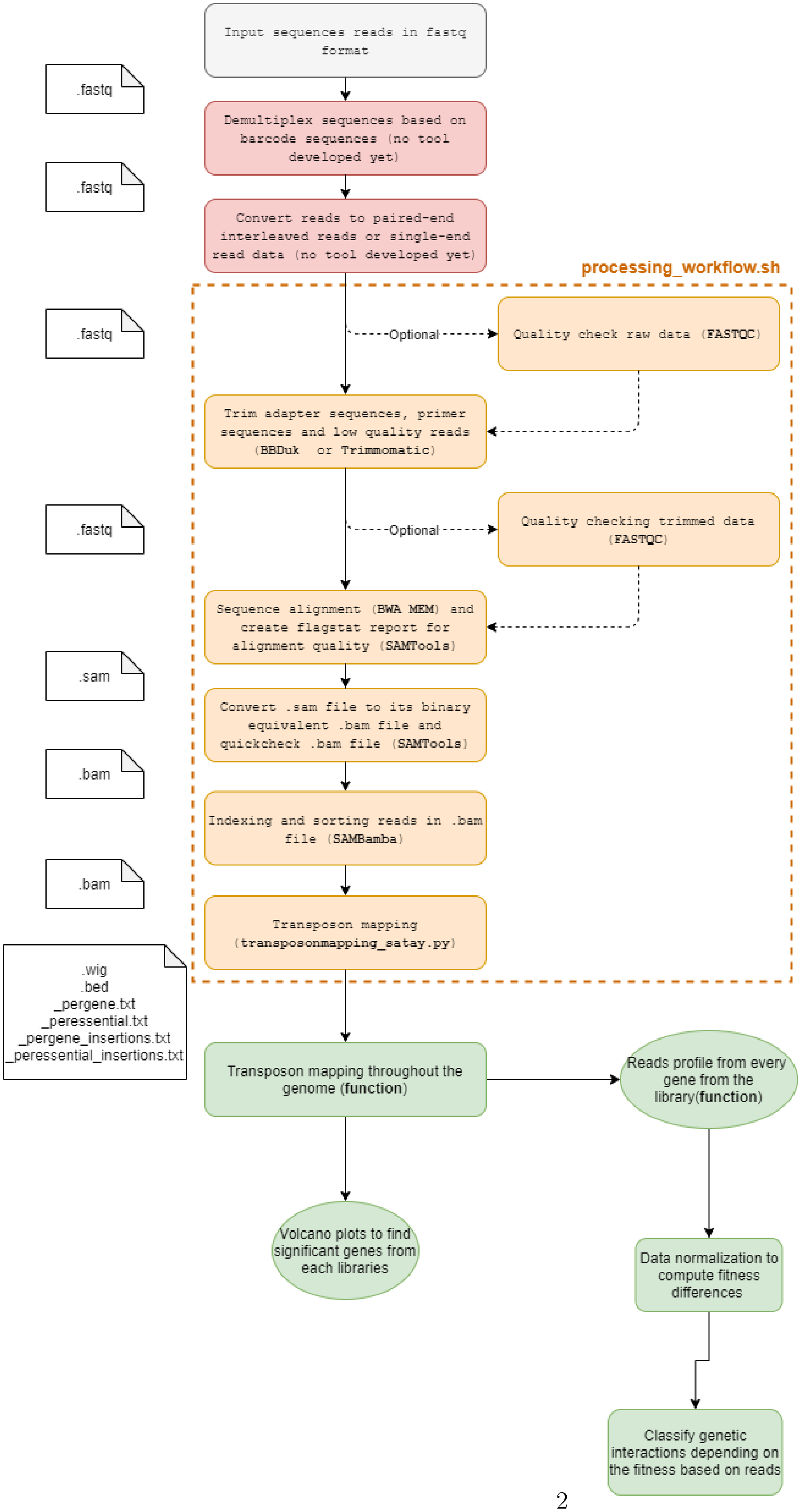
The program can trim sequencing reads, create quality reports of raw data and of the trimmed data, align the reads to a reference genome (S288C yeast genome, downloaded from the SGD) in .sam- and .bam-format, sort and index the bam file and perform transposon-mapping. The program can be ran interactively using a GUI or it can be used with command line arguments.

The GUI is automatically started when running the satay Docker container without passing any arguments.

The following tools are used in the processing pipeline:

- quality report: FASTQC
- trimming: BBDuk
- alignment: BWA MEM
- create flagstat report after alignment: SAMTools
- sort and index .bam-files: SAMBamba
- transposon mapping: https://github.com/SATAY-LL/Transposonmapper/blob/main/transposonmapper/transposonmapper.py

Processing pipeline


Input sequences reads in fastq format
.fastq
Demultiplex sequences based on barcode sequences (no tool developed yet)
.fastq
Convert reads to paired-end interleaved reads or single-end read data (no tool developed yet)
processing_workflow.sh
.fastq
Optional
Quality check raw data (FASTQC)
Trim adapter sequences, primer sequences and low quality reads (BBDuk or Trimmomatic)
.fastq
Optional
Quality checking trimmed data (FASTQC)
Sequence alignment (BWA MEM) and create flagstat report for alignment quality (SAMTools)
.sam
Convert .sam file to its binary equivalent .bam file and quickcheck .bam file (SAMTools)
.bam
Indexing and sorting reads in .bam file (SAMBamba)
.bam
Transposon mapping (transposonmapping_satay.py)
.wig
.bed
_pergene.txt
_peressential.txt
_pergene_insertions.txt
_peressential_insertions.txt
Transposon mapping throughout the genome (function)
Reads profile from every gene from the library(function)
Volcano plots to find significant genes from each libraries
Data normalization to compute fitness differences
Classify genetic interactions depending on the fitness based on reads

First time use

In order to trim the dataset, you need to provide the adapters sequence.

Find adapters sequence

A way to obtain the adapters sequence, is to run the pipeline with the following optins checked

☒ Quality checking raw data
☒ Quality check interrupt

Then:

1. When the GUI asks you to continue, say NO (the container will stop) and go to your local `/data/fastqc_out/` folder
2. Open the corresponding html file (on your own system) and go to the "Overrepresented sequences" section
3. Copy the sequence that has more than 15% of representation.

Create the adapterfile.fa in your local data folder

1. `cd /data`
2. `nano adapterfile.fa`
3. Edit the file as follows:
   NAME
   Paste the copied sequence
4. Ctrl-O to save the file, Ctrl-X to quit the editor

Rerun the container

With the new adapters.fa file present in your data folder, rerun the container. The software will notify you it will use your custom adapters.fa file

GUI

Data input and output

When no command line arguments are given, the program launched the GUI which start with a window where the datafile(s) can be selected. The datafiles should be in fastq format and must have the extension .fastq or .fq. They can be either unpacked or gzipped (i.e. having the extension .gz). Select the right extension in the bottom right corner and navigate to the datafile(s). In case of single-end data or paired-end interleaved data select only one file. In case of paired-end data where the pairs are stored in two seperate files, select two files by holding ctrl-button and clicking the two files. After pressing `ok`, a new window appears where some options and parameters can be selected.

GUI input fields

- **file primary reads:** Show the selected file(s).
- **file secondary reads:** Show the selected file(s). If only one file is chosen, the file secondary reads will show `none`.
- **Data type:** Select whether the reads are paired-end or single-end. If two data files were chosen but this setting is set to Single-end, the secondary reads file will be ignored.

- **Trimming software:** Select whether to use `bbduk` or select `donottrim` to prevent trimming of the reads.
- **Trimming settings for bbduk**: Input trimming settings, see the documentation of the selected trimming software which settings can be applied.
  - When the trimming software is set to `donottrim`, this field will be ignored.
  - Sequences that need to trimmed (e.g. adapter or primer sequences) have to be entered in the adapters.fa file which can be accessed using the `Open adapters file` button on the bottom of the window.
  - **bbduk**: ktrim=l k=15 mink=10 hdist=1 qtrim=r trimq=10 minlen=30 [https://jgi.doe.gov/data-and-tools/bbtools/bb-tools-user-guide/bbduk-guide/] Note: For bbduk do not input `interleaved=t` when using interleaved data. For trimmomatic do not input `SE` or `PE` to indicate single-end or paired-end data. This will all be automatically set depending on your selection in the `Data type`-field.
  - (Deprecated) Trimmmomatic: ILLUMINACLIPPING:1:30:10 TRAILING:20 SLIDINGWINDOW:5:10 MINLEN:15 [http://www.usadellab.org/cms/uploads/supplementary/Trimmomatic/TrimmomaticManual_V0.32.pdf] Note: For trimmomatic, when using ILLUMINACLIP, do not specify the path to the adapters file as this is inserted automatically (see example settings below).
- **Alignment settings:** See the documentation of BWA MEM which settings can be applied. Note: Do not set `-p` for smart pairing (i.e. interleaved paired-end data), this will be automatically set depending on your selection in the 'Data type'-field. [http://bio-bwa.sourceforge.net/bwa.shtml]

Processing steps checkboxes

- **Quality checking raw data:** Perform a fastqc quality check on the raw reads.
- **Quality checking trimmed data:** Perform a fastqc quality check on the trimmed reads. This setting is ignored if 'which trimming to use' it set to `donottrim`.
- **Quality check interrupt:** This quits the program after performing the quality report on the raw dataset and creating a temporary file with your settings. This allows you to check the quality report before continuing. To continue the program, restart the program (using bash processing_workflow.sh). It will automatically set the options you have chosen the first time, but these can be changed if this is necessary depending on the outcome of the quality report. This option can be useful if you have no idea how the dataset looks. This requires 'Quality checking raw data'.
- **Delete sam file:** After alignment the .sam file is converted to its binary equivalent and only this .bam file is used for downstream processing. Since the .sam file typically requires a lot of memory, this is can be deleted. It is recommended to keep the .sam file only for manual checking te alignment results.
- **Sort and index bam files:** This is needed for transposon-mapping and for many other downstream processes. It is recommended to always leave this on.
- **Transposon mapping:** This custom python script requires sorting and indexing of the .bam file and creates the following files:
  1. .bed file: Creates list of all insertion locations with the number of reads in each location in bed-format.
  2. .wig file: Creates list of all insertion locations with the number of reads in each location in wig-format. Small difference with the bed-file is that here reads from insertions at the same location but with different orientation are added up. In the bed-file these are regarded two separate insertions.
  3. .txt-files: List of all genes with the number of insertions and reads in each gene. The files are

different in whether they show all genes or all annotated essential genes and whether they also show the distribution of insertions within the genes.

- **Create flagstat report:** Creates a flagstat report based on the .bam file.

Buttons

- **Open adapters file**: Opens the text file where the adapter and primer sequences can be entered that will be trimmed. Enter the sequences in fasta format.
- **Quick reference guide:** Open the documentation in the container. For full documentation, please visit https://satay-ll.github.io/SATAY-jupyter-book/Introduction.html

Command line usage

For using the program with the command line, the following arguments can be passed:

- `bash $satay -ARGS`
- [-h] Show help text
- [-v] Show current version
- [-c] Open the adapters file. This does not run the program
- [-f] Select data file with primary reads (required)
- [-g] Select data file with secondary reads (only in case of paired-end noninterleaved data)
- [-p] Select data format. Either 'Paired-end' or 'Single-end' [default is 'Single-end']
- [-s] Select which trimming software to use. Either 'bbduk', 'trimmomatic' or 'donottrim' (use the latter to skip trimming) [default is 'bbduk']
- [-t] Input trimming options (preferably use ") [default is 'ktrim=l k=15 mink=10 hdist=1 qtrim=r trimq=10 minlen=30' which is used for bbduk].
- [-a] Input alignment options (preferably use ") [default is '-v 2']
- [-i] Run index-and-sorting of bam-file [TRUE or FALSE, default is TRUE]
- [-m] Run transposon mapping [TRUE or FALSE, default is TRUE]
- [-d] Delete .sam file [TRUE or FALSE, default is TRUE]
- [-q] Quality report for samflags [TRUE or FALSE, default is TRUE]
- [-x] Quality report raw sequencing data [TRUE or FALSE, default is FALSE]
- [-y] Quality report trimmed sequencing data [TRUE or FALSE, default is TRUE]
- [-z] Interrupt program after raw sequencing quality report [TRUE or FALSE, default is FALSE]

For questions

For questions, recommendations and issues can be noted at https://github.com/SATAY-LL/Transposonmapper/issues/new/choose For more detailed information about this program see https://satay-ll.github.io/SATAY-jupyter-book/Introduction.html