SomatoSim User Manual

Marwan A. Hawari, Celine S. Hong and Leslie G. Biesecker

Contact: marwan.hawari@nih.gov or celine.hong@nih.gov

November 10, 2020

Contents

1 Introduction

1.1 Software dependencies

1. Python version 3.6.8 (https://www.python.org/downloads/)
2. Numpy version 1.16.2 (https://numpy.org/)
3. Pandas version 0.25.1 (https://pandas.pydata.org/)
4. Matplotlib version 3.1.1 (https://matplotlib.org/)
5. Pysam version 0.15.0 (https://pysam.readthedocs.io/en/latest/)
6. SAMtools version 1.9 (http://www.htslib.org/)

1.2 Download

Visit https://github.com/BieseckerLab/SomatoSim for scripts and test data.

1.3 Installation

Clone the github repository and execute the following commands:

```
$ git clone https://github.com/BieseckerLab/SomatoSim.git
$ cd SomatoSim
$ python3 -m pip install .
```

1.4 Attribution

This is software was developed by Marwan A. Hawari, Celine S. Hong, and Leslie G. Biesecker at the National Human Genome Research Institute (NHGRI), National Institutes of Health (NIH). Please include proper attribution of the NHGRI as the developer of this program and include a link to the following [https://github.com/BieseckerLab/SomatoSim] in all publications and other public disclosures that reference the program and/or include data or research results that were generated utilizing the program.

1.5 Public Domain Notice

This software is a United States Government Work. Anyone may use the software on a worldwide and royalty-free basis for any purpose and anyone may reproduce and prepare derivative works without restriction. Although all reasonable efforts have been taken to ensure the accuracy and reliability of the software, the National Human Genome Research Institute (NHGRI), National Institutes of Health (NIH) and the U.S. Government do not and cannot warrant the performance or any results that may be obtained by using this software. NHGRI, NIH and the U.S. Government disclaim all warranties as to performance, merchantability or fitness for any particular purpose. No indemnification is intended or provided by the US government.

2 Usage

2.1 Command line execution

```
$ somatosim \
  -i <input bam file path> \
  -b <input bed file path> \
  -o <output directory path> \
  --vaf-low <number> \
  --vaf-high <number> \
  --output-prefix <output files prefix> \
  --number-snv <number> \
  --random-seed <number> \
  --down-sample <number> \
  --target-coverage <number> \
  --coverage-tolerance <number> \
  --coverage-MQ <number> \
  --coverage-BQ <number> \
  --minimum-separation <number> \
  <--verbose> \
  --read-min-MQ <number> \
  --position-min-BQ <number>
```

2.2 Command line options descriptions

-h, --help
Show the SomatoSim help message and exit

--version
Show the SomatoSim version number and exit

Required arguments:

-i, --input-bam-file
The full path of the input BAM file. See Section 3.1 for details.

-b, --input-bed-file
The full path of the input BED file. See Section 3.2 for details.

-o, --output-directory
The full path where the SomatoSim output files will be deposited. If the directory does not already exist, it will automatically be created.

Optional arguments:

--vaf-low
The lower bound of the desired VAF range. Input this value as a decimal, not as a percentage. This is ignored if the BED file input contains pre-defined VAF values. [Default: 0.01]

--vaf-high

The upper bound of the desired VAF range. Input this value as a decimal, not as a percentage. This is ignored if the BED file input contains pre-defined VAF values. [Default: 0.20]

--output-prefix

The desired prefix for the output files generated by SomatoSim. [Default: input BAM file prefix]

--number-snv

The desired number of SNVs to simulate. [Default: Number of lines in the input BED file]

--random-seed

The desired random seed. Choosing a seed will help with reproducibility. [Default: random number between 0 and 100]

--down-sample

The desired average coverage to which the input BAM file will be down-sampled. [Default: No down-sampling]

--target-coverage

The desired depth of coverage to target when selecting for genomic positions. [Default: None]

--coverage-tolerance

The tolerance allowed when targeting a specific depth of coverage. Input this value as a decimal, not as a percentage. This option is ignored if the --target-coverage value is not used. [Default: 0.10]

--coverage-MQ

The mapping quality used by SAMtools mpileup when calculating coverage. [Default: 20]

--coverage-BQ

The base quality used by SAMtools mpileup when calculating coverage. [Default: 20]

--minimum-separation

Set the minimum number of bases that must separate simulated variants. For example, if this value is 0, then variants can be simulated directly adjacent to each other. [Default: 1]

--verbose

Displays the specific positions not selected during variant selection stage. Typing --verbose will turn the option on, and not typing anything will leave it off. [Default: Off]

--read-min-MQ

The minimum mapping quality used when selecting reads to introduce simulated variants into. [Default: 20]

--position-min-BQ

The minimum base quality used when selected reads to introduce simulated variants into. [Default: 20]

2.3 Run time considerations

Table 1. Summary of how each input of SomatoSim can affect the run time

Argument	Change	Run time	Rationale
`-b/--input-bam-file`	Increase BAM file size	↑	More time necessary to compute average coverage and iterate through the reads
`--vaf-low`	Increase value	↑	This will result in more reads needing to be selected and mutated
`--vaf-high`	Increase value	↑	This will result in more reads needing to be selected and mutated
`--number-snv`	Increase value	↑	Selecting for more positions and reads means searching longer for those positions and reads
`--random-seed`	Change value	—	Changing random seed only changes random values
`--down-sample`	Selecting a value	↑	SomatoSim needs to down-sample the BAM file
`--target-coverage`	Selecting a value	↑	SomatoSim needs to iterate through more bases to select bases at the target coverage
`--coverage-tolerance`	Increase value	↓	Increasing the coverage tolerance allows more positions to be selected in each loop
`--coverage-MQ`	Change value	—	Changes the SAMtools parameter when the depth of coverage is calculated
`--coverage-BQ`	Change value	—	Changes the SAMtools parameter when the depth of coverage is calculated
`--minimum-separation`	Increase value	↑	Increasing the minimum number of bases in between spike-ins may make selecting suitable positions more difficult
`--read-min-MQ`	Increase value	↑	Increasing the minimum read MQ may make selecting suitable reads more difficult
`--position-min-BQ`	Increase value	↑	Increasing the minimum BQ for a position may make selecting suitable reads more difficult

3 Input

3.1 BAM file

The BAM file input for SomatoSim should be processed and analysis ready. This means that any pre-processing practices such as marking duplicates or base quality score recalibration should be completed prior to using SomatoSim for simulating SNVs. Additionally, the input BAM file should contain only paired end reads and be karyotypically sorted and indexed.

3.2 BED file

Random variant selection

With random variant selection, the input BED file needs only three fields: the chromosome, the genomic range start position, and the genomic range end position.

```
1 14042036 14042109
1 63048855 63048925
1 53796840 53796923
1 156127861 156127923
```

User-specified variant selection

With user-specified variant selection, the input BED file can contain specific single positions, VAF values, and variant alleles.

If the user wishes to simulate a variant at a very specific single genomic position, they can do so using the 0-based genomic coordinate format. In this format, the VAF and variant allele will still be randomly assigned to each position.

```
1 63048906 63048907
1 53796891 53796892
1 156127917 156127918
1 92729200 92729201
```

If you are specifying *only* specific VAF values, those values should be in the fourth field:

```
1 63048906 63048907 0.06
1 53796891 53796892 0.06
1 156127917 156127918 0.03
1 92729200 92729201 0.14
```

If specifying *only* specific variant alleles, those values should be in the fourth field:

```
1 63048906 63048907 G
1 53796891 53796892 T
1 156127917 156127918 G
1 92729200 92729201 G
```

If specifying *both* specific VAF values and variant alleles, the VAF values should be in the fourth field and the variant alleles should be in the fifth field, or vice versa.

```
1 63048906 63048907 0.06 G
1 53796891 53796892 0.06 T
1 156127917 156127918 0.03 G
1 92729200 92729201 0.14 G
```

4 Output

1. Simulated BAM file
A file with ".somatosim.bam" suffix and the associated ".somatosim.bam.bai" index file. The BAM file contains the mutated reads, while the original BAM file is unchanged.

2. Simulation output text file
A text file in a 0-based BED format containing information on the final output of the simulation. These are the final positions that were successfully mutated. The file has the following fields:

```
[chromosome/ position/ position/ input_VAF/ input_coverage/
output_VAF/ output_coverage/ ref_allele/ alt_allele/]
```

3. Simulation log file
This log file contains metrics from the variant selection stage, variant simulation stage, and the variant evaluation stage. Notably, the log file will report the user's input parameters, the version numbers of SomatoSim and its dependencies, the run time, the number of positions selected for variant simulation, the number of reads altered, the final number of positions where variants were simulated, the VAF distributions, and the variant allele distributions among other metrics.

4. Simulation failed mutation text file
A text file in a 0-based BED format containing positions with failed mutations. This file is not generated if there are no failed mutations. The file has the following fields:

```
[chromosome/ position/ position/ input_VAF/ coverage/
variant_coverage/ count_T_in/ count_G_in/ count_A_in/ count_C_in/
count_total_in/ monoallelic_in/ ref_allele/ alt_allele/ count_T_out/
count_G_out/ count_A_out/ count_C_out/ count_total_out/]
```

5 Test data

The test data BAM file is derived from the National Institute of Standards and Technology (NIST) Genome in a Bottle (GIAB) NA12878 HiSeq 300X BAM file (Zook *et al*., 2016).

https://www.nist.gov/programs-projects/genome-bottle

ftp://ftp-trace.ncbi.nlm.nih.gov/giab/ftp/data/NA12878/NIST_NA12878_HG001_HiSeq_300x/NHGRI_Illumina300X_novoalign_bams/HG001.hs37d5.300x.bam

The exon regions used in the test BED file were derived from the GENCODE Release 27 (GRCh37) comprehensive gene annotation gff3 file (Frankish *et al*., 2019).

https://www.gencodegenes.org/human/release_27lift37.html

ftp://ftp.ebi.ac.uk/pub/databases/gencode/Gencode_human/release_27/GRCh37_mapping/gencode.v27lift37.annotation.gff3.gz

To create the test BED file, 12 exonic regions for each chromosome (including both X and Y) were randomly selected from the GENCODE exon annotation. These selected exonic regions were intersected with the GIAB BAM file to create a sub-sampled BAM file known to contain the exonic regions in the test BED file. The advantage of this is that high depth of coverage in the original GIAB data was preserved but the BAM file size was reduced by several orders of magnitude. The BAM file was then re-aligned to the hs37d5 reference genome using SAMtools fastq and BWA-mem and had duplicate reads marked by Picard MarkDuplicates.

The test_BED_user.bed file was created by randomly selecting a single genomic position from each genomic range in the BED file and assigning it a VAF and variant allele.

6 Tutorial

1. Download and install dependencies:

Make sure you have the latest version of SAMtools and pip installed.
Refer to the SAMtools documentation (http://www.htslib.org/download/) for SAMtools installation instructions. When you install SomatoSim, pip will also automatically download and install any python packages you need to use SomatoSim. You can upgrade your current version of pip using the following command:

```
$ python3 -m pip install --upgrade pip
```

2. Download and install SomatoSim:

```
$ git clone https://github.com/BieseckerLab/SomatoSim.git
$ cd SomatoSim
$ python3 -m pip install .
```

3. Run SomatoSim:

Here we will simulate 100 SNVs with VAFs between 0.01 and 0.05. The input BAM and BED files are supplied in this SomatoSim repository.

```
$ somatosim \
  -i test_data/test_BAM.bam \
  -b test_data/test_BED.bed \
  -o ./output_dir \
  --vaf-low 0.01 \
  --vaf-high 0.05 \
  --number-snv 100 \
  --random-seed 0
```

4. Explore SomatoSim's outputs

The output_dir directory should contain 4 files:

- A new BAM file: `test_BAM.somatosim.bam`
- A new BAM index file: `test_BAM.somatosim.bam.bai`
- A results file: `simulation_output.txt`
- A log file: `simulation_log.txt`

Below is an example of the first 10 lines of the results file. The first 3 columns are the SNV coordinates, the input_VAF column is the target VAF, the input_coverage column is the original coverage at the position, the output_VAF column is the VAF calculated after making the mutations, the output_coverage column is the coverage after making the mutations (and should be the same as the `input_VAF` column), the `ref_allele` column is the original allele at the position, and the `alt_allele` is the new allele introduced by SomatoSim.

```
chromosome position position input_VAF input_coverage output_VAF output_coverage ref_allele alt_allele
1          109898015   109898016   0.05        329         0.04863     329         C           A
1          211571650   211571651   0.05        285         0.04912     285         C           T
1          14042085    14042086    0.04        380         0.03947     380         C           T
1          156127891   156127892   0.05        269         0.04833     269         G           T
1          198222215   198222216   0.05        399         0.05013     399         G           C
1          197009494   197009495   0.02        335         0.0209      335         A           G
1          153148814   153148815   0.01        289         0.01038     289         T           A
1          92729264    92729265    0.03        425         0.03059     425         T           G
2          89999382    89999383    0.03        242         0.02893     242         T           G
```

Below is an example of the log file. It reports which versions of the dependencies are currently in use, the user's input parameters, and metrics related to the three stages (variant selection, variant simulation, and variant evaluation) of SomatoSim.

```
  ____                        _        ____  _
 / ___|  ___  _ __ ___   __ _| |_ ___ / ___|(_)_ __ ___
 \___ \ / _ \| '_ ` _ \ / _` | __/ _ \\___ \| | '_ ` _ \
  ___) | (_) | | | | | | (_| | || (_) |___) | | | | | | |
 |____/ \___/|_| |_| |_|\__,_|\__\___/|____/|_|_| |_| |_|

Mon Oct 26 16:43:06 2020

SomatoSim version 1.0.0
Python version 3.6.10
Numpy version 1.19.2
Pandas version 1.1.3
Matplotlib version 3.3.2
Pysam version 0.16.0.1
Samtools version 1.11

Log file for: test_BAM

Simulation run parameters:
--------------------------
BAM file path: test_data/test_BAM.bam
BED file path: test_data/test_BED.bed
BED file input contains no pre-defined VAF or alternate allele

VAF range: 0.01 - 0.05
Output directory: ./output_dir
Using default output prefix
Number of positions to mutate: 100
Random seed: 0
Samtools mpileup coverage calculation using mapping quality (MQ) = 20 and base quality (BQ) =
20
Minimum base pair separation = 1
Verbose mode: Off

Variant selection stage set-up:
-------------------------------
Using input BED format

Average coverage of original BAM file: 289.784


Variant selection stage metrics:
--------------------------------
Total number of genomic positions in the BED file: 159709
Number of genomic ranges in the BED file: 288
```

```
Total number of genomic positions iterated through: 137
Final number of input positions selected to mutate: 100
Average coverage of the final selected input positions: 284.2

VAF distribution (input):
[[ 0.    9.  ]
 [ 0.01 20.  ]
 [ 0.02 17.  ]
 [ 0.03 26.  ]
 [ 0.04 20.  ]
 [ 0.05  8.  ]
 [ 0.06  0.  ]]

Alternate allele distribution (input):
[['T' '26']
 ['G' '27']
 ['A' '24']
 ['C' '23']]

Variant selection stage runtime: 9.046 seconds

Variant simulation stage metrics:
---------------------------------
Minimum mapping quality for reads to mutate: 20
Minimum base quality for variant allele: 20

Number of mutations expected from the variant selection stage: 880
Number of reads selected for variant simulation: 880
Total number of mutations: 880
Total reads in the input BAM file: 341240

Variant simulation stage runtime: 44.823 seconds

Variant evaluation stage metrics:
-------------------------------
Final number of unique spike-in positions inside the target VAF value range: 100
Final mean coverage at spike-in positions inside the target VAF value range: 284.2

VAF distribution (output):
[[ 0.    9.  ]
 [ 0.01 20.  ]
 [ 0.02 17.  ]
 [ 0.03 26.  ]
 [ 0.04 20.  ]
 [ 0.05  8.  ]
 [ 0.06  0.  ]]

Alternate allele distribution (output):
[['T' '26']
 ['G' '27']
 ['A' '24']
 ['C' '23']]

####################
Simulation complete!
####################

Variant evaluation stage runtime: 1.814 seconds

Total runtime: 55.684 seconds
```

7 References

Frankish,A. *et al.* (2019). GENCODE reference annotation for the human and mouse genomes. *Nucleic Acids Res.,* **47**, D766-D773.

Zook,J.M. *et al.* (2016). Extensive sequencing of seven human genomes to characterize benchmark reference materials. *Sci Data,* **3**, 160025.