SCSit: A high-efficiency preprocessing tool for single-cell sequencing data from SPLiT-seq

**Mei-Wei Luan1,#, Jia-Lun Lin2,#, Ye-Fan Wang1, Yu-Xiao Liu2, Chuan-Le Xiao3, Rongling Wu4, Shang-Qian Xie1,\***

\* To whom correspondence should be addressed:

Shang-Qian Xie, Email: [sqianxie@foxmail.com](mailto:sqianxie@foxmail.com)

July10, 2021

**Contents**

[**1.** **Introduction** 1](#_Toc74844682)

[**2.** **Basic Operations** 1](#_Toc74844683)

[**2.1** **Installation** 1](#_Toc74844684)

[**2.2** **Getting started** 1](#_Toc74844685)

[**2.3** **SCSit output** 2](#_Toc74844686)

[**3.** **Citation and Contact** 2](#_Toc74844687)

1. **Introduction**

**What is SCSit**

SPLiT-seq provides a low-cost platform to generate single-cell data by labeling the cellular origin of RNA through four rounds of combinatorial barcoding. Before analysing this sequence to draw biological conclusions you should always automatically and rapidly perform preprocess and quality control from SPLiT-seq, which directly identified and labeled combinatorial barcoding reads.

SCSit can directly identify combinatorial barcodes and UMI of cell types and obtain more labeled reads, and remarkably enhance the retained data from SCS due to the exact alignment of insertion and deletion. The consistency of identified reads from SCSit increases to 97%, and mapped reads are twice than the original alignment method (e.g. BLAST and BWA). Furthermore, the runtime of SCSit is less than 10% of the original. It can accurately and rapidly analyze SPLiT-seq raw data and obtain labeled reads, as well as effectively improve the single-cell data from SPLiT-seq platform. The data and source of SCSit are available on the GitHub website <https://github.com/shang-qian/SCSit>.

1. **Basic Operations**
   1. **Installation**

SCSit is part of the Anaconda distribution and can be installed with Anaconda or Miniconda in PyPI.

|  |
| --- |
| pip install -i https://pypi.anaconda.org/susiew01/simple scsit-tools |

Or,

|  |
| --- |
| conda install –c susiew01 scsit |

Or,

SCSit is optimized for x86-64 CPUs. You can acquire precompiled C program from the release page with:

|  |
| --- |
| git clone <https://github.com/shang-qian/SCSit.git>  cd SCSit  ./make.sh |

* 1. **Getting started**

SCSit takes a pair of pair-end sequencing read files from SPLiT-seq in FASTQ format, barcodeList and primerList in TXT format, and outputs a pair of files in FASTQ format. SCSit only supports files in the following formats FastQ (all quality encoding variants).

Newly opened files will immediately appear in finished percentage of running files. Because of the size of these files it can take a couple of minutes to open them. SCSit operates a pair of files multithreading at a time.

Example:

|  |
| --- |
| scsit -r1 input\_r1.fastq -r2 input\_r2.fastq -p primer.list -b barcode.list -t 4 -o output |

**Required arguments:**

-r1 R1File : Paths to files that contain input read1 of SPLiT-seq pair-end files

-r2 R2File : Paths to files that contain input read2 of SPLiT-seq pair-end files

-p primerList : Primer list of all oligonucleotide sequences used

-b barcodeList : The 96 well plate oligonucleotides used for each round of barcodes

**Optional arguments:**

-o output prefix : Paths to files that contain output file prefix

-t int : Maximum number of threads to use [default: 2]

-h : show this help message and exit

* 1. **SCSit output**

The raw reads FASTQ format of SPLiT-seq data as input executed SCSit program and output labeled reads with combinatorial barcodes and UMI. The output Read1 FASTQ format file was composed of combinatorial barcodes and UMI, and Read2 file was corresponding sequencing data.

1. **Citation and Contact**

MW, Luan. et al. SCSit: A high-efficiency preprocessing tool for single-cell sequencing data from SPLiT-seq.

Shang-Qian Xie, Email: sqianxie@foxmail.com