Manual of NCLscan

Description

We have developed a new pipeline, NCLscan, which is rather advantageous in the identification of "non-co-linear" (NCL) transcripts (fusion, *trans*-splicing, and circular RNA) from paired-end RNA-seq data.

The NCLscan program, document, and test set can be downloaded from our FTP site: ftp://treeslab1.genomics.sinica.edu.tw/NCLscan or GitHub: https://github.com/TreesLab/NCLscan.

1. System requirements

NCLscan runs under the Linux (i.e., Bio-Linux, see also http://nebc.nerc.ac.uk/ for further details) or Mac operating systems on a 64-bit machine with ≥ 10 GB RAM.

2. Software requirements

The following five tools are involved in the NCLscan pipeline:
(1) Bedtools (http://bedtools.readthedocs.org/en/latest/)
(2) Samtools (http://www.htslib.org/)
(3) BLAT (https://genome.ucsc.edu)
(4) BWA (http://bio-bwa.sourceforge.net/)
(5) Novoalign, novoindex, and novoalign license file (http://www.novocraft.com/)

These executable files of bedtools, samtools, blat, bwa, novoalign, novoindex, and novoalign.lic should be copied to the file directory of NCLscan.

Note: Novoalign is a commercial bioinformatics tool. It can be requested for a limited license for academic/non-profit researches.

3. Installation

The users can download NCLscan_v1.3.tar.gz file and unzip it.
> tar xvzf NCLscan_v1.3.tar.gz

3.1 Reference preparation

The genomic sequences and annotation were downloaded from the GENCODE website at http://www.gencodegenes.org/. Given the human reference genome (GRCh37.p13, http://www.gencodegenes.org/releases/19.html) as an example, the following four reference files are required:

(1) Genome sequence FASTA file in GRCh37.p13 assembly (sequence region names are the same as in the GTFs): GRCh37.p13.genome.fa.gz
(2) Protein-coding transcript sequences: gencode.v19.pc_transcripts.fa.gz
(3) Long non-coding RNAs: gencode.v19.lncRNA_transcripts.fa.gz
(4) Gene annotation: gencode.v19.annotation.gtf.gz

Unzip these 4 reference files, and move them into the file directory of NCLscan.

The human reference files could also be downloaded from our FTP website (ftp://treeslab1.genomics.sinica.edu.tw/NCLscan/NCLscan_reference)

```
> tar jxvf NCLscan_reference_hg19.tar.bz2
> cd /path/to/NCLscan_reference_hg19
> mv . /path/to/NCLscan_v1.3
```

3.2 Configuration

The above four reference files and two parameters are listed in "config.txt" as follows:

```
====================
GRCh37.p13.genome.fa
gencode.v19.pc_transcripts.fa
gencode.v19.lncRNA_transcripts.fa
gencode.v19.annotation.gtf
151
500
====================
```

Note: The final two parameters, 151 and 500, indicate the maximal read length (L) and fragment size of the used paired-ended RNA-seq data (FASTQ files), where fragment size = 2*L + insert size.
If L > 151, the users should change these two parameters to (L, 2*L + insert size).

3.3 BWA index preparation

Before executing BWA alignment, the users need to create mapping index files for the reference genome. For the human reference genome (GRCh37.p13), the index files ("bwa_AllRef.fa*") has been created and accessible at our FTP site (ftp://treeslab1.genomics.sinica.edu.tw/NCLsacn/BWAindex/).

>tar jxvf NCLscan_bwa_index_hg19.tar.bz2

> cd /path/to/NCLscan_bwa_index_hg19

> mv . /path/to/NCLscan_v1.3

The users can also generate index ("bwa_AllRef.fa*") by themselves using the following command:

> cd /path/to/NCLscan

> ./MakeBwaRef

4. Execution of NCLscan

Usage:

>./NCLscan.sh 01.fastq 02.fastq MyProject 20 8 50

Note: 01.fastq and 02.fastq are the two files of a set of paired-end RNA-seq data. NCLscan outputs two main results: "MyProject.result" and "MyProject.result.sam". MyProject is the prefix of the output filenames. The fourth parameter means the cutoff of sequence quality score of the considered RNA-seq reads (default: 20). If a read contains a base with sequence quality score that is smaller than the cutoff value, such a read is not considered. If 0 is set, it means that all reads are considered. The fifth parameter is the number of used threads for BWA (default: 8). The sixth parameter is the size of span range (default: 50; see section 5).

An example of executing NCLscan:

A test RNA-seq dataset "simu_5X_100PE" is provided in the test folder.

Step 1. Create a test project folder

> mkdir test_NCLscan

Step 2. The shortcuts of NCLscan_v1.3 are created in the test_project folder.

> cd test_NCLscan

> ln -s /path/to/NCLscan_v1.3/* .

Step3. Run it

> ./NCLscan.sh /path/to/simu_5X_100PE_1.fastq /path/to/simu_5X_100PE_2.fastq test_NCLscan 0 8 50 2>&1 | tee test_NCLscan.log

The final result will be generated as a “test_NCLscan.result” file.

5. NCLscan outputs

Two output files are generated after executing the pipeline: "MyProject.result" (tab-delimited text file) and “MyProject.result.sam” (SAM-format file). The former is the list of the identified NCL events, and the latter records the supporting reads of the identified NCL events. For the SAM format, users can see http://samtools.github.io/hts-specs/SAMv1.pdf for further details.

The column format of "MyProject.result" is described as follows:
(1) Chromosome name of the donor side (5'ss)
(2) Junction coordinate of the donor side
(3) Strand of the donor side
(4) Chromosome name of the acceptor side (3'ss)
(5) Junction coordinate of the acceptor side
(6) Strand of the acceptor side
(7) Gene name of the donor side
(8) Gene name of the acceptor side
(9) Intragenic (1) or intergenic (0) case

6. The detailed description of the NCLscan pipeline

The NCLscan pipeline includes six steps, which are all involved in the batch file (NCLscan.sh). The six steps and their usages are explained as follows:
Step 1: NCL_Scan0
Usage:
>./NCL_Scan0.sh 01.fastq 02.fastq MyProject

Note: NCL_Scan0 aligns the reads against the reference genome and the annotated transcripts using BWA.

Step 2: NCL_Scan1:
Usage:
> NCL_Scan1 MyProject.bwa.unmapped_1.fastq MyProject.bwa.unmapped_2.fastq

Note: NCL_Scan1 aligns the BWA-unmapped reads against the reference genome and the annotated transcripts using Novoalign.

Step 3: NCL_Scan2

Usage:

> NCL_Scan2 20

Note: NCL_Scan2 includes three steps: (1) concatenating the two ends of each unmapped read; (2) BLAT-aligning the concatenated sequences against the reference genome; and (3) removing the concatenated sequences with an alternative co-linear explanation. If a read contains a base with sequence quality score that is smaller than the cutoff value (default: 20), such a read is not considered.

Step 4: NCL_Scan3

Usage:

> NCL_Scan3

Note: NCL_Scan3 generates putative NCL references with putative NCL junction sites for the retained concatenated-sequences.

Step 5: NCL_Scan4

Usage:

> NCL_Scan4 50

Note: NCL_Scan4 includes the following steps:

1. aligning unmapped reads against all the putative NCL references using Novoalign,
2. retaining the putative NCL references that satisfy all of the three rules: (1) all mapped reads must not have any alternative co-linear explanation; (2) at least one read supports the junction site; and (3) the collection of the supported reads must span the NCL junction boundary by ≥the setting size of span range (default: 50 bp) on both sides of the junction site,
3. aligning the retrieved putative NCL references against the reference genome with a different set of BLAT parameters
4. removing candidates with an alternative co-linear explanation within a single gene or between-in two close genes.

Step 6: NCL_Scan5

Usage:

>./NCL_Scan5.py

Note: This program appends gene names in the final result (i.e., "MyProject.result") according to the gene annotation.