SpoTyping

SpoTyping is a software for predicting spoligotype from sequencing reads, complete genomic sequences and assembled contigs.

Part I. Spoligotype prediction and SITVIT database query.

Prerequisites:

- Python2.7
- BLAST

Input:

1. Fastq file or pair-end fastq files
2. Fasta file of a complete genomic sequence or assembled contigs of an isolate

Output:

In the output file specified: predicted spoligotype in the format of binary code and octal code.
In the output log file: count of hits from BLAST result for each spacer sequence.
In the xls excel file: spoligotype query result downloaded from SITVIT WEB.

- Note: if the same spoligotype is queried before and have an xls file in the output directory, it will not be queried again.

Usage:

python SpoTyping.py [options] FASTQ_1 FASTQ_2(optional)

An Example call:
python2.7 SpoTyping.py read_1.fastq read_2.fastq –o spo.out

Options:

--version
show program's version number and exit

-h, --help
show this help message and exit

--seq
Set this if input is a fasta file that contains only complete genomic sequence or assembled contigs from an isolate. [Default is off]

-s SWIFT, --swift=SWIFT
swift mode, either "on" or "off" [Default: on]

-m MIN, --min=MIN
minimum number of error-free hits to support presence of a spacer [Default: 5]

-r MIN_RELAX, --rmin=MIN_RELAX
minimum number of 1-error-tolerant hits to support presence of a spacer [Default: 6]

-O OUTDIR, --outdir=OUTDIR
output directory [Default: running directory]

-o OUTPUT, --output=OUTPUT
basename of output files generated [Default: SpoTyping]

--noQuery
suppress the SITVIT database query [Default is off]

-d, --debug
enable debug mode, keeping all intermediate files for checking [Default is off]

FASTQ_1/FASTA
input FASTQ read 1 file or sequence fasta file (mandatory)

FASTQ_2
input FASTQ read 2 file (optional for pair-end reads)

Suggestions:

1. It's highly suggested to use the swift mode (set as the default) if the sequencing throughput is no less than 135Mbp.
2. For sequencing experiments with throughputs below 135Mbp, please adjust the thresholds to be 0.0180 to 0.1486 times the estimated read depth for error-free hits and 0.0180 to 0.1488 times the estimated read depth for 1-error-tolerant hits. (The read depth is estimated by dividing the sequencing throughput by 4,500,000, which is the estimated *Mtb* genome length.)
3. If you do wish to take in all reads for sequencing experiments with throughputs above 1260Mbp, please adjust the thresholds to be 0.0180 to 0.1486 times the estimated read depth for error-free hits and 0.0180 to 0.1488 times the estimated read depth for 1-error-tolerant hits.

Got weird spoligotype prediction?

- **Sequencing throughput is very high** (>1000Mbp, for example): try to disable the swift mode and set the hit thresholds higher (10% of the estimated read depth, for example).

```
# Example commad:
python SpoTyping.py -s off -m 10 -r 12 read_1.fastq.gz read_2.fastq.gz
```

- **Sequencing throughput is very low** (<40Mbp, for example): SpoTyping may not be able to give accurate prediction due to the relatively low read depth.

Part II. Summary pie chart plot from the downloaded xls files.

Prerequisites:

- R
- R package: gdata

Input:

The xls file downloaded from SITVIT WEB.

Output:

A pdf file with the information in the xls file summarized with pie charts.

Usage:

Rscript SpoTyping_plot.r query_from_SITVIT.xls output.pdf

An example call:

Rscript SpoTypint_plot.r SITVIT_ONLINE.777777477760771.xls SITVIT_ONLINE.777777477760771.pdf