tngs

(perl edition)

version 2022.09.16

Introduction

The tngs project aims to provide a collection of tools supporting work with microbial genomics. This is the perl edition which is designed to provide a collection of useful command line tools in a single script with few dependencies. A Java version with a graphical user interface is also under construction (not publicly available yet).

To run the script, a perl interpreter is needed which is usually present on linux and mac-os but on windows it may be necessary to install it manually (e.g. strawberryperl or activeperl).

Installation

The script is ready to use, if a perl interpreter is available. Make sure the script has permission to be executable. You might want to place the script in a location covered by the $PATH. (e.g. for Ubuntu: home/youruser/bin (make the bin dir if it is not present and make a log-out/log-in again)

Parameter convention

The script must be given a number of parameters. If the script is run without parameters it displays the tngs version and a list of available tools. The first parameter is always the tool. The following parameters will depend on the tool selected. Three types of parameters are used. "Flags" are parameters turning on a specific feature and always begin with "--" (e.g., . "--verbose"). Parameters with values are always directly followed by a "=" and then the value. No spaces are allowed surrounding the "=" (e.g., "kmersize=30"). The remaining parameters that are not flags (no "--") and not value-parameters (no "=") are usually interpreted as input filenames, directorynames or paths.
The names of flags and parameters with values are case insensitive, while the values and the remaining parameters are case sensitive.
If parameters are given, that are not expected by the tool, a warning is usually printed.

Input files and output files can always either be uncompressed or compressed with gzip. If they end with ".gz", the gzip algorithm is invoked. The compression/uncompression is made by perl IO:Zlib and gzip is not required (e.g. when used on windows).

Tools

The current tools/commands are:

version
help
fileinfo
downsample
filtercontigs
kmeroverlapp
kmerhisto

version

The “version” command prints the current version of tngs.

Aliases: version, -version, --version, -v

Flags:
“--verbose” Will print versions of all tools and sub-tools.

help

The “help” command prints a list of the available tools/commands. This is the same message printed when tngs is run without parameters.

fileinfo

The “fileinfo” tool first determines the file type by looking at the file extension and then passes on the appropriate subtool. Alternatively a directory can be given and then all files in the directory will be analyzed. If the “--recursive” flag is used, all subdirectories are also scanned (recursively). If the input is a directory, either the “--fasta” flag or “--fastq” flag must be specified, to determine what type of scan that is to be made.

fileinfo (subtool fastq)

The analysis of fastq files assumes the fastq file contains short read data (typically Illumina data). If the data is from a paired end sequencing run, the default behaviour of the program is to try to identify both the R1 and the R2 read files and co-analyze them. Thus, only one file must be specified (either the R1 file or the R2 file). To override this behaviour, the flag “--unpaired” can be used. Then the fastq file is analyzed without searching for a paired file.

Running tngs fileinfo on a fastq file will generate some basic quantifications and QC metrics. These include:

Nr reads / read pairs
Nr bases
Nr bases >=Q30
Nrbases <=Q20
GC content
Number of ‘N’ bases
Range of and average sequence length

The first 100k reads are analyzed a little bit deeper (limited for speed reasons)

The best matching 16S sequences found are quantified and reported (done for selected species only).
Matches to Illumina adapter sequences are quantified and reported.

Species that are searched for are:

Campylobacter jejuni/coli
Listeria monocytogenes
Escherichia coli
Mycobacterium tuberculosis
Klebsiella pneumoniae
Streptococcus pneumoniae
Staphylococcus aureus

If a species is identified with a sufficient number of reads (default 0.1% reads hitting the 16S sequence), some more metrics are given:

Coverage
Coverage only counting Q30+ bases

Flags:

--tabular will output the information in a tab separated table format (default when directory is scanned)
--notabular output in text format
--noheader will skip the header row in tabular output format (may be used to get an output format that can be appended to a table with ">>" in linux systems)
--verbose will output more information (also non discriminating 16S hits)
--unpaired Do not search for and coanalyze with a paired fastq file
--recursive Search also subdirectories recursively (subdirs, sub-subdirs …etc)

Extensions recognized as fastq files are

.fastq
.fq
.fastq.gz
.fq.gz

fileinfo (subtool fasta)

The analysis of fasta files assumes the fasta file contains a genome or a genome assembly.

Running tngs fileinfo on a fasta file will generate some basic quantifications and assembly QC metrics. These include:

Nr sequences (contigs)
Nr bases
GC content and base composition
Number of 'N' bases
Range of and average sequence length
N50
L50

N90
L90
Number contigs >=500
Assembly size with contigs >=500
Number contigs >=1kb
Assembly size with contigs >=1kb

If the assembly contig naming is in SPAdes format, the number of low coverage contigs are quantified. A low coverage contig has less than 10% of the average k-mer coverage of the assembly and usually comes from a contamination in the WGS analysis. The following info is reported:

Number of low coverage contigs
Cumulative size of low coverage contigs
Cumulative size of low coverage contigs expressed as percent of assembly size
Size of the assembly if the low coverage contigs were to be filtered out
Average k-mer coverage for this assembly, reported by SPAdes

The best matching 16S sequences found are quantified and reported (done for selected species only).
If present, matches to Illumina adapter sequences are reported.

Species that are searched for are:

Campylobacter jejuni/coli
Listeria monocytogenes
Escherichia coli
Mycobacterium tuberculosis
Klebsiella pneumoniae
Streptococcus pneumoniae
Staphylococcus aureus

Flags:

--tabular will output the information in a tab separated table format
--notabular do not use tabular format (which is default when used on a directory)
--noheader will skip the header row in tabular output format (may be used to get an output format that can be appended to a table with “>>” in linux systems)
--recursive will search subdirs recursively

File extensions:

File extensions recognized as fasta are:

.fasta
.fa
.fna

The extensions can also have a .gz ending.

downsample

The “downsample” tool takes one (or two if paired end) fastq files and information about the wanted amount of bases and then downsamples the fastq file(s) to the target size.

The wanted amount of bases can be specified in several ways:

bases=100000000
number of bases should be 100000000

coverage=100 genomesize=4000000
number of bases should give 100X coverage when the genomesize is 4 Mb (400000000 bases).

coverage=50 organism=campylobacter
Number of bases should give 50X coverage when the genome size is like a typical campylobacter genome.

Available organisms:

organism=campylobacter (1.7 Mb)
organism=salmonella (5 Mb)
organism=listeria (3 Mb)
organism=ecoli (5 Mb)
organism=mycobacterium_tuberculosis (4.4 Mb)
organism=klebsiella_pneumoniae (5.7 Mb)
organism=streptococcus_pneumoniae (2.15 Mb)
organism=staphylococcus_aureus (2.8 Mb)

The output file(s) will by default be named the same as the inputfile(s) but with a “.d” inserted before the extension (e.g., myfile_R1.d.fastq.gz)

If custom names are wanted use

out1=newname_1.fastq.gz out2=newname_2.fastq.gz

Flags:

--q30bases   count only bases with quality Q30 or higher.

filtercontigs

The “filtercontigs” tool takes a WGS assembly (fasta file) and removes short (<200 bp) and low-coverage contigs (<10% of average assembly k-mer coverage). Note!, Low-coverage filtering contig is only available when SPAdes style contig names are provided.

The thresholds can be adjusted

sizethreshold=500
coveragethreshold=20

The output file will be named the same as the input file but with a ".f." before the extension (e.g., myfile.f.fasta".

Flags:

--verbose output detailed information about each contig (size and coverage, keep or filter)

kmeroverlapp

The "kmeroverlapp" tool takes two genomes/WGS assemblies and compares how many of the kmers (default=20-mers) in one of the genomes are present in the other (and the other way around). The k-mer size can be changed e.g. kmersize=19.
The default behaviour is that all k-mers are compared. Alternatively, only unique k-mers can be compared (use the .--unique flag). In that case, repetitive k-mers are ignored in the comparison.

Flags:

--unique only compare unique k-mers (ignore repetitive k-mers)

kmerhisto

The "kmerhisto" tools takes one (or two if paired end) fastq files and makes a histogram of the frequencies of the k-mers present (default 20-mers). The k-mer size can be changed e.g.

kmersize=19.

Optionally a reference genome sequence can be given, and then the histogram will show the frequencies in the fastq files of the k-mers present in the reference genome.

reference=myreference.fasta

The histogram has by default 500 bins. This can be changed:

bins=300

To get comparable histograms from different samples, a specific amount of bases may be used from the fastq files:

maxbases=50000000

The analysis will produce a table with the histogram values that can be plotted with a graph producing software. It will also try to estimate the genomesize based on the coverage peak position in comparison to the amount of data used.

Optionally, a GC-bias table can also be produced. This is a table that prints out the average GC-content of every coverage bin. This can be used to detect if GC-content affects coverage. This function can be activated by adding the "--gcbiasplot" flag.