Supplementary Methods

Compressor selection

The current study benchmarks compressors that can compress DNA/RNA/protein sequences in FASTA format, without using reference sequence. We used all specialized sequence compressors that we could find and make to work. For general-purpose compressors we used only the major ones, in terms of performance, historical importance, or popularity.

We did not benchmark any compressors that:

- Work only with FASTQ data
- Perform lossy compression
- Require reference sequence
- Compress only aligned sequences
- Require commercial license for academic use
- Don't have command line interface
- Can't be obtained anymore
- Are distributed only as Windows binaries

For each compressor that has parameters related to compression strength of speed, we used the relevant range of settings, including the default.

Benchmark machine

- CPU: dual Xeon E5-2643v3 (3.4GHz, 6 cores)
- Hyperthreading: off
- RAM: 128 GB DDR4-2133 ECC Registered
- Storage: 4 x 2 TB SSD, in RAID 0
- OS: Ubuntu 18.04.1 LTS

Test procedure

Only one compression or decompression task was running at a time. The machine was not performing any other work in parallel with the benchmark.

For both compression and decompression, the input file was first loaded into disk cache of the OS, to remove the influence of the disk access speed.

For decompression runs, first the verification run was executed, where the decompressed data was saved to disk, and then verified for identity with the original data (using md5sum). Only compressors and settings that produced decompressed data byte-to-byte identical with the original were included in benchmark.

After successful verification, the actual decompression run was performed and timed. In this run, the decompressed data was piped to /dev/null, in order to remove the influence of storing the data.

Any compression or decompression run that completed in under 1 second was repeated 10 times, and any run that completed in under 3 seconds was repeated 3 times. The shortest of the measured times was recorded. This was done to achieve at least minimal rejection of noise in time measurements.

Compressor setup

For compression, each compressor was reading the input data streamed via pipe ("|" in the command line). For decompression, each compressor was set up to stream the decompressed data via pipe. This was done to better approximate the most likely pattern of using a compressor in actual sequence compression application. In an actual sequence analysis workflow, often the decompressed data is piped directly into the downstream analysis command. Also, when compressing the sequences, often the data is first pre-processed with another command, which then pipes the processed sequences to the compressor.

Since some compressors do not support such streaming mode of operation, we used them via wrapper scripts. Our wrapper scripts also fix other deficiencies of many compressors, including:

- Supporting RNA input for DNA-only compressors.
- Supporting 'N' in DNA/RNA sequences.
- Supporting IUPAC's ambiguous nucleotide codes.
- Saving and restoring line lengths.
- Saving and restoring sequence names.
- Saving and restoring sequence mask (upper/lower case).
- Supporting FASTA-formatted input.
- Supporting input with more than 1 sequence.

All our wrappers and commands are available at the Benchmark Database Website (http://kirr.dyndns.org/sequence-compression-benchmark/).