
seqmagick Documentation

Release 0.8.6

Matsen Group

Jun 20, 2023

Contents

1	Changes for seqmagick	3
2	Motivation	7
3	Installation	9
4	Use	11
5	List of Subcommands	13
6	Supported File Extensions	21
7	Acknowledgements	23
8	Contributing	25

Contents

- *seqmagick*
 - *Motivation*
 - *Installation*
 - *Use*
 - *List of Subcommands*
 - *Supported File Extensions*
 - * *Default Format*
 - * *Compressed file support*
 - *Acknowledgements*
 - *Contributing*

Changes for seqmagick

1.1 0.8.0

- Supports Python 3.5+
- Drops support for Python 3.4
- Fix issue: “seqmagick with no params gives KeyError:None” [GH-77]
- Fix for Biopython 1.71 dual coding support [GH-76]; also fixes issue: “Translation error with new BioPython” [GH-79]
- Send logging to stderr, not stdout [GH-75]

1.2 0.7.0

- Supports Python 3.4+
- Drops support for python 2.7
- requires biopython >= 1.70
- Drops support for bz2 compression [see GH-66]
- New option `convert --sample-seed` to make `--sample` deterministic.

1.3 0.6.2

- New `quality-filter --pct-ambiguous` switch [GH-53]
- `setup.py` enforces `biopython>=1.58,<=1.66` (1.67 is not compatible) [GH-59]
- This is the last release that will support Python 2!

1.4 0.6.1

- Allow string wrapping when input isn't FASTA. [GH-45]
- Fix `--pattern-include`, `--pattern-exclude`, and `--pattern-replace` for sequences without descriptions (e.g., from NEXUS files). [GH-47]
- Fix mogrify example. [GH-52]

1.5 0.6.0

- Map `.nex` extension to NEXUS-format (`--alphabet` must be specified if writing)
- Use reservoir sampling in `--sample` selector (lower memory use)
- Support specifying negative indices to `--cut` [GH-33]
- Optionally allow invalid codons in `backtrans-align` [GH-34]
- Map `.fq` extension to FASTQ format
- Optional multithreaded I/O in `info` [GH-36]
- Print sequence name on length mismatch in `backtrans-align` [GH-37]
- Support for `+` and `-` in head and tail to mimic Linux head and tail commands.
- Fix scoring for mixed-case sequences in `primer-trim`.
- Fix bug in `primer-trim` - failed when sequence had multiple 5' gaps compared to the primer.
- Clarify documentation and fix bug in `convert/mogrify --pattern-replace` [GH-39]
- Support for gzip files in `seqmagick convert --sort`

1.6 0.5.0

- Change `seqmagick extract-ids --source-format` to `--input-format` to match other commands (GH-29)
- Support gzip- and bzip2-compressed inputs and outputs for most commands (GH-30)
- Change default input format for `sff` to `sff-trim`, which respects the clipping locations embedded in each sequence record.
- Add `--details-out` option to `seqmagick quality-filter`, which writes details on each read processed.
- Match barcode/primer `seqmagick quality-filter` against a trie; allows per-specimen barcodes.
- Remove `--failure-out` option from `seqmagick quality-filter`. See `--details-out`
- Raise an error if number of codons does not match number of amino acids in `seqmagick backtrans-align`
- Add `--sample` subcommand (GH-31)

1.7 0.4.0

- Fix bug in `--squeeze`
- More informative messages in `seqmagick primer-trim`
- Added `--alphabet` flag to allow writing NEXUS (GH-23)
- Exiting without error on SIGPIPE in `extract-ids`, `info` (GH-17)
- Ambiguities are translated as 'X' in `-translate` (GH-16)
- Allowing '.' or '-' as gap character (GH-18)
- `--name-prefix` and `--name-suffix` no longer create a mangled description (GH-19)
- Files owned by another user can be mogrified, as long as they are group writeable (GH-14)
- Add `backtrans-align` subcommand, which maps unaligned nucleotides onto a protein alignment (GH-20)
- Allow FASTQ as input to `quality-filter`
- Significantly expand functionality of `quality-filter`: identify and trim barcodes/primers; report detailed failure information.
- Cleanup, additional tests
- Add `--drop` filter to `convert` and `mogrify` (GH-24)
- Apply current umask when creating files (GH-26)
- Support stdin in `seqmagick info` (GH-27)
- Support translating ambiguous nucleotides, if codon translation is unambiguous

1.8 0.3.1

- Fix bug in `quality-filter MinLengthFilter`
- Case consistency in `seqmagick`

1.9 0.3.0

- Internal reorganization - transformations are converted to partial functions, then applied.
- Argument order now affects order of transformation application.
- Change default output format to 'align' for TTYs in `seqmagick info`
- Add BioPython as dependency (closes GH-7)
- Add `primer-trim` subcommand
- Add option to apply custom function(s) to sequences
- Add new filtering options: `--squeeze-threshold`, `--min-ungapped-length`, `--include-from-file` `--exclude-from-file`
- Removed `seqmagick muscle`
- Added new subcommand `quality-filter`

- Added new subcommand `extract-ids` (closes GH-13)
- Allow use of '-' to indicate stdin / stdout (closes GH-11)
- Add mapping from `.phyx` to `phylip-relaxed` (targeted for BioPython 1.58)

1.10 0.2.0

- Refactoring
- Added hyphenation to multi-word command line options (e.g. `--deduplicatetaxa` -> `--deduplicate-taxa`)
- Add support for `.needle`, `.sff` formats
- Close GH-4

1.11 0.1.0

Initial release

CHAPTER 2

Motivation

We often have to convert between sequence formats and do little tasks on them, and it's not worth writing scripts for that. Seqmagick is a kickass little utility built in the spirit of [imagemagick](#) to expose the file format conversion in Biopython in a convenient way. Instead of having a big mess of scripts, there is one that takes arguments:

```
seqmagick convert a.fasta b.phy      # convert from fasta to phylip
seqmagick mogrify --ungap a.fasta    # remove all gaps from a.fasta, in place
seqmagick info *.fasta               # describe all FASTA files in the current directory
```

And more.

CHAPTER 3

Installation

Install the latest release with:

```
pip install seqmagick
```

This should also install [BioPython](#). NumPy (which parts of BioPython depend on) is not required for seqmagick to function, but may be installed as a dependency of BioPython.

To install the bleeding edge version:

```
pip install git+https://github.com/fhcrc/seqmagick.git@master#egg-info=seqmagick
```

Note that as of version 0.8.0, this package requires Python 3.5+. If you want to use the most recent version compatible with Python 2.7:

```
pip install seqmagick==0.6.2
```


CHAPTER 4

Use

Seqmagick can be used to query information about sequence files, convert between types, and modify sequence files. All functions are accessed through subcommands:

```
seqmagick <subcommand> [options] arguments
```

List of Subcommands

5.1 convert and mogrify

Convert and mogrify achieve similar goals. `convert` performs some operation on a file (from changing format to something more complicated) and writes to a new file. `mogrify` modifies a file in place, and would not normally be used to convert formats.

The two have similar signatures:

```
seqmagick convert [options] infile outfile
```

vs:

```
seqmagick mogrify [options] infile
```

Options are shared between `convert` and `mogrify`.

5.1.1 Examples

Basic Conversion

`convert` can be used to convert between any file types BioPython supports (which is many). For a full list of supported types, see the [BioPython SeqIO wiki page](#).

By default, file type is inferred from file extension, so:

```
seqmagick convert a.fasta a.sto
```

converts an existing file `a.fasta` from FASTA to Stockholm format. **Neat!** But there's more.

Sequence Modification

A wealth of options await you when you're ready to do something slightly more complicated with your sequences.

Let's say I just want a few of my sequences:

```
$ seqmagick convert --head 5 examples/test.fasta examples/test.head.fasta
$ seqmagick info examples/test*.fasta
name                alignment  min_len  max_len  avg_len  num_seqs
examples/test.fasta FALSE      972     9719    1573.67   15
examples/test.head.fasta FALSE      978      990     984.00    5
```

Or I want to remove any gaps, reverse complement, select the last 5 sequences, and remove any duplicates from an alignment in place:

```
seqmagick mogrify --tail 5 --reverse-complement --ungap --deduplicate-sequences_
↳examples/test.fasta
```

You can even define your own functions in python and use them via `--apply-function`.

Note: To maximize flexibility, most transformations passed as options to `mogrify` and `convert` are processed *in order*, so:

```
seqmagick convert --min-length 50 --cut 1:5 a.fasta b.fasta
```

will work fine, but:

```
seqmagick convert --cut 1:5 --min-length 50 a.fasta b.fasta
```

will never return records, since the cutting transformation happens before the minimum length predicate is applied.

Command-line Arguments

```
usage: seqmagick convert [-h] [--line-wrap N]
                        [--sort {length-asc,length-desc,name-asc,name-desc}]
                        [--apply-function /path/to/module.py:function_]
↳name[:parameter]]
                        [--cut start:end[,start2:end2]] [--relative-to ID]
                        [--drop start:end[,start2:end2]] [--dash-gap]
                        [--lower] [--mask start1:end1[,start2:end2]]
                        [--reverse] [--reverse-complement] [--squeeze]
                        [--squeeze-threshold PROP]
                        [--transcribe {dna2rna,rna2dna}]
                        [--translate {dna2protein,rna2protein,dna2proteinstop,
↳rna2proteinstop}]
                        [--ungap] [--upper] [--deduplicate-sequences]
                        [--deduplicated-sequences-file FILE]
                        [--deduplicate-taxa] [--exclude-from-file FILE]
                        [--include-from-file FILE] [--head N]
                        [--max-length N] [--min-length N]
                        [--min-ungapped-length N] [--pattern-include REGEX]
                        [--pattern-exclude REGEX] [--prune-empty]
                        [--sample N] [--sample-seed N]
                        [--seq-pattern-include REGEX]
```

(continues on next page)

(continued from previous page)

```

[--seq-pattern-exclude REGEX] [--tail N]
[--first-name] [--name-suffix SUFFIX]
[--name-prefix PREFIX]
[--pattern-replace search_pattern replace_pattern]
[--strip-range] [--input-format FORMAT]
[--output-format FORMAT]
[--alphabet {dna,dna-ambiguous,rna,rna-ambiguous,protein}]
source_file dest_file

```

Convert between sequence formats

positional arguments:

```

source_file      Input sequence file
dest_file        Output file

```

optional arguments:

```

-h, --help          show this help message and exit
--alphabet {dna,dna-ambiguous,rna,rna-ambiguous,protein}
                    Input alphabet. Required for writing NEXUS.

```

Sequence File Modification:

```

--line-wrap N       Adjust line wrap for sequence strings. When N is 0,
                    all line breaks are removed. Only fasta files are
                    supported for the output format.
--sort {length-asc,length-desc,name-asc,name-desc}
                    Perform sorting by length or name, ascending or
                    descending. ASCII sorting is performed for names

```

Sequence Modification:

```

--apply-function /path/to/module.py:function_name[:parameter]
                    Specify a custom function to apply to the input
                    sequences, specified as
                    /path/to/file.py:function_name. Function should accept
                    an iterable of Bio.SeqRecord objects, and yield
                    SeqRecords. If the parameter is specified, it will be
                    passed as a string as the second argument to the
                    function. Specify more than one to chain.
--cut start:end[,start2:end2]
                    Keep only the residues within the 1-indexed start and
                    end positions specified, : separated. Includes last
                    item. Start or end can be left unspecified to indicate
                    start/end of sequence. A negative start may be
                    provided to indicate an offset from the end of the
                    sequence. Note that to prevent negative numbers being
                    interpreted as flags, this should be written with an
                    equals sign between '--cut' and the argument, e.g.:
                    '--cut=-10:'
--relative-to ID     Apply --cut relative to the indexes of non-gap
                    residues in sequence identified by ID
--drop start:end[,start2:end2]
                    Remove the residues at the specified indices. Same
                    format as '--cut'.
--dash-gap           Replace any of the characters "?::~" with a "-" for
                    all sequences
--lower             Translate the sequences to lower case
--mask start1:end1[,start2:end2]
                    Replace residues in 1-indexed slice with gap-

```

(continues on next page)

(continued from previous page)

```

characters. If --relative-to is also specified,
coordinates are relative to the sequence ID provided.
--reverse          Reverse the order of sites in sequences
--reverse-complement  Convert sequences into reverse complements
--squeeze          Remove any gaps that are present in the same position
                   across all sequences in an alignment (equivalent to
                   --squeeze-threshold=1.0)
--squeeze-threshold PROP
                   Trim columns from an alignment which have gaps in
                   least the specified proportion of sequences.
--transcribe {dna2rna,rna2dna}
                   Transcription and back transcription for generic DNA
                   and RNA. Source sequences must be the correct alphabet
                   or this action will likely produce incorrect results.
--translate {dna2protein,rna2protein,dna2proteinstop,rna2proteinstop}
                   Translate from generic DNA/RNA to proteins. Options
                   with "stop" suffix will NOT translate through stop
                   codons . Source sequences must be the correct alphabet
                   or this action will likely produce incorrect results.
--ungap           Remove gaps in the sequence alignment
--upper           Translate the sequences to upper case

```

Record Selection:

```

--deduplicate-sequences
                   Remove any duplicate sequences by sequence content,
                   keep the first instance seen
--deduplicated-sequences-file FILE
                   Write all of the deduplicated sequences to a file
--deduplicate-taxa   Remove any duplicate sequences by ID, keep the first
                   instance seen
--exclude-from-file FILE
                   Filter sequences, removing those sequence IDs in the
                   specified file
--include-from-file FILE
                   Filter sequences, keeping only those sequence IDs in
                   the specified file
--head N            Trim down to top N sequences. With the leading `-',
                   print all but the last N sequences.
--max-length N      Discard any sequences beyond the specified maximum
                   length. This operation occurs *before* all length-
                   changing options such as cut and squeeze.
--min-length N      Discard any sequences less than the specified minimum
                   length. This operation occurs *before* cut and
                   squeeze.
--min-ungapped-length N
                   Discard any sequences less than the specified minimum
                   length, excluding gaps. This operation occurs *before*
                   cut and squeeze.
--pattern-include REGEX
                   Filter the sequences by regular expression in ID or
                   description
--pattern-exclude REGEX
                   Filter the sequences by regular expression in ID or
                   description
--prune-empty       Prune sequences containing only gaps ('-')
--sample N          Select a random sampling of sequences
--sample-seed N     Set random seed for sampling of sequences

```

(continues on next page)

(continued from previous page)

```
--seq-pattern-include REGEX
                        Filter the sequences by regular expression in sequence
--seq-pattern-exclude REGEX
                        Filter the sequences by regular expression in sequence
--tail N                Trim down to bottom N sequences. Use +N to output
                        sequences starting with the Nth.
```

Sequence ID Modification:

```
--first-name            Take only the first whitespace-delimited word as the
                        name of the sequence
--name-suffix SUFFIX    Append a suffix to all IDs.
--name-prefix PREFIX    Insert a prefix for all IDs.
--pattern-replace search_pattern replace_pattern
                        Replace regex pattern "search_pattern" with
                        "replace_pattern" in sequence ID and description
--strip-range           Strip ranges from sequences IDs, matching </x-y>
```

Format Options:

```
--input-format FORMAT   Input file format (default: determine from extension)
--output-format FORMAT  Output file format (default: determine from extension)
```

Filters using regular expressions are case-sensitive by default. Append "(?i)" to a pattern to make it case-insensitive.

5.2 backtrans-align

Given a protein alignment and unaligned nucleotides, align the nucleotides using the protein alignment. Protein and nucleotide sequence files must contain the same number of sequences, in the same order, with the same IDs.

```
usage: seqmagick backtrans-align [-h] [-o destination_file]
                                [-t {standard,standard-ambiguous,vertebrate-mito}]
                                [-a {fail,warn,none}]
                                protein_align nucl_align
```

Given a protein alignment **and** unaligned nucleotides, align the nucleotides using the protein alignment. Protein **and** nucleotide sequence files must contain the same number of sequences, **in** the same order, **with** the same IDs.

positional arguments:

```
protein_align    Protein Alignment
nucl_align       FASTA Alignment
```

optional arguments:

```
-h, --help            show this help message and exit
-o destination_file, --out-file destination_file
                        Output destination. Default: STDOUT
-t {standard,standard-ambiguous,vertebrate-mito}, --translation-table {standard,
↪ standard-ambiguous,vertebrate-mito}
                        Translation table to use. [Default: standard-
                        ambiguous]
-a {fail,warn,none}, --fail-action {fail,warn,none}
                        Action to take on an ambiguous codon [default: fail]
```

5.3 extract-ids

seqmagick extract-ids is extremely simple - all the IDs from a sequence file are printed to stdout (by default) or the file of your choosing:

```
usage: seqmagick extract-ids [-h] [-o OUTPUT_FILE]
                             [--input-format INPUT_FORMAT] [-d]
                             sequence_file

Extract the sequence IDs from a file

positional arguments:
  sequence_file          Sequence file

optional arguments:
  -h, --help              show this help message and exit
  -o OUTPUT_FILE, --output-file OUTPUT_FILE
                          Destination file
  --input-format INPUT_FORMAT
                          Input format for sequence file
  -d, --include-description
                          Include the sequence description in output [default:
                          False]
```

5.4 info

seqmagick info describes one or more sequence files

5.4.1 Example

```
seqmagick info examples/*.fasta
```

name	alignment	min_len	max_len	avg_len	num_seqs
examples/aligned.fasta	TRUE	9797	9797	9797.00	15
examples/dewrapped.fasta	TRUE	240	240	240.00	148
examples/ range .fasta	TRUE	119	119	119.00	2
examples/test.fasta	FALSE	972	9719	1573.67	15
examples/wrapped.fasta	FALSE	120	237	178.50	2

Output can be in comma-separated, tab-separated, or aligned formats. See seqmagick info -h for details.

Usage:

```
usage: seqmagick info [-h] [--input-format INPUT_FORMAT]
                     [--out-file destination_file] [--format {tab,csv,align}]
                     [--threads THREADS]
                     sequence_files [sequence_files ...]

Info action

positional arguments:
  sequence_files
```

(continues on next page)

(continued from previous page)

```
optional arguments:
  -h, --help                show this help message and exit
  --input-format INPUT_FORMAT
                           Input format. Overrides extension for all input files
  --out-file destination_file
                           Output destination. Default: STDOUT
  --format {tab,csv,align}
                           Specify output format as tab-delimited, CSV or aligned
                           in a borderless table. Default is tab-delimited if the
                           output is directed to a file, aligned if output to the
                           console.
  --threads THREADS        Number of threads (CPUs). [1]
```

5.5 quality-filter

quality-filter truncates and removes sequences that don't match a set of quality criteria. The subcommand takes a FASTA and quality score file, and writes the results to an output file:

```
usage: seqmagick quality-filter [-h] [--input-qual INPUT_QUAL]
                                [--report-out REPORT_OUT]
                                [--details-out DETAILS_OUT]
                                [--no-details-comment]
                                [--min-mean-quality QUALITY]
                                [--min-length LENGTH] [--max-length LENGTH]
                                [--quality-window-mean-qual QUALITY_WINDOW_MEAN_QUAL]
                                [--quality-window-prop QUALITY_WINDOW_PROP]
                                [--quality-window WINDOW_SIZE]
                                [--ambiguous-action {truncate,drop}]
                                [--max-ambiguous MAX_AMBIGUOUS]
                                [--pct-ambiguous PCT_AMBIGUOUS]
                                [--primer PRIMER | --no-primer]
                                [--barcode-file BARCODE_FILE]
                                [--barcode-header] [--map-out SAMPLE_MAP]
                                [--quoting {QUOTE_ALL,QUOTE_MINIMAL,QUOTE_NONE,QUOTE_
↪NONNUMERIC}]
                                sequence_file output_file
```

Filter reads based on quality scores

```
positional arguments:
  sequence_file            Input fastq file. A fasta-format file may also be
                           provided if --input-qual is also specified.
  output_file              Output file. Format determined from extension.

optional arguments:
  -h, --help                show this help message and exit
  --input-qual INPUT_QUAL
                           The quality scores associated with the input file.
                           Only used if input file is fasta.
  --min-mean-quality QUALITY
                           Minimum mean quality score for each read [default:
                           25.0]
  --min-length LENGTH      Minimum length to keep sequence [default: 200]
  --max-length LENGTH      Maximum length to keep before truncating [default:
```

(continues on next page)

(continued from previous page)

```

        1000]. This operation occurs before --max-ambiguous
--ambiguous-action {truncate,drop}
        Action to take on ambiguous base in sequence (N's).
        [default: no action]
--max-ambiguous MAX_AMBIGUOUS
        Maximum number of ambiguous bases in a sequence.
        Sequences exceeding this count will be removed.
--pct-ambiguous PCT_AMBIGUOUS
        Maximun percent of ambiguous bases in a sequence.
        Sequences exceeding this percent will be removed.

Output:
--report-out REPORT_OUT
        Output file for report [default: stdout]
--details-out DETAILS_OUT
        Output file to report fate of each sequence
--no-details-comment
        Do not write comment lines with version and call to
        start --details-out

Quality window options:
--quality-window-mean-qual QUALITY_WINDOW_MEAN_QUAL
        Minimum quality score within the window defined by
        --quality-window. [default: same as --min-mean-
        quality]
--quality-window-prop QUALITY_WINDOW_PROP
        Proportion of reads within quality window to that must
        pass filter. Floats are [default: 1.0]
--quality-window WINDOW_SIZE
        Window size for truncating sequences. When set to a
        non-zero value, sequences are truncated where the mean
        mean quality within the window drops below --min-mean-
        quality. [default: 0]

Barcode/Primer:
--primer PRIMER
        IUPAC ambiguous primer to require
--no-primer
        Do not use a primer.
--barcode-file BARCODE_FILE
        CSV file containing sample_id,barcode[,primer] in the
        rows. A single primer for all sequences may be
        specified with '--primer', or '--no-primer' may be
        used to indicate barcodes should be used without a
        primer check.
--barcode-header
        Barcodes have a header row [default: False]
--map-out SAMPLE_MAP
        Path to write sequence_id,sample_id pairs
--quoting {QUOTE_ALL,QUOTE_MINIMAL,QUOTE_NONE,QUOTE_NONNUMERIC}
        A string naming an attribute of the csv module
        defining the quoting behavior for `SAMPLE_MAP`.
        [default: QUOTE_MINIMAL]

```

Supported File Extensions

By default, `seqmagick` infers the file type from extension. Currently mapped extensions are:

Extension	Format
.afa	fasta
.aln	clustal
.fa	fasta
.faa	fasta
.fas	fasta
.fasta	fasta
.fastq	fastq
.ffn	fasta
.fna	fasta
.fq	fastq
.frn	fasta
.gb	genbank
.gbk	genbank
.needle	emboss
.nex	nexus
.phy	phylip
.phylip	phylip
.phyx	phylip-relaxed
.qual	qual
.sff	sff-trim
.sth	stockholm
.sto	stockholm

Note: NEXUS-format output requires the `--alphabet` flag.

6.1 Default Format

When reading from stdin or writing to stdout, `seqmagick` defaults to fasta format. This behavior may be overridden with the `--input-format` and `--output-format` flags.

If an extension is not listed, you can either rename the file to a supported extension, or specify it manually via `--input-format` or `--output-format`.

6.2 Compressed file support

most commands support gzip (files ending in `.gz`) and bzip (files ending in `.bz2` or `.bz`) compressed inputs and outputs. File types for these files are inferred using the extension of the file after stripping the file extension indicating that the file is compressed, so `input.fasta.gz` would be inferred to be in FASTA format.

CHAPTER 7

Acknowledgements

seqmagick is written and maintained by the [Matsen Group](#) at the Fred Hutchinson Cancer Research Center.

CHAPTER 8

Contributing

We welcome contributions! Simply fork the repository on [GitHub](#) and send a pull request.