CIAlign

Release 1.1.0

Charlotte Tumescheit, Andrew Firth, Katy Brown

Apr 12, 2023

CONTENTS

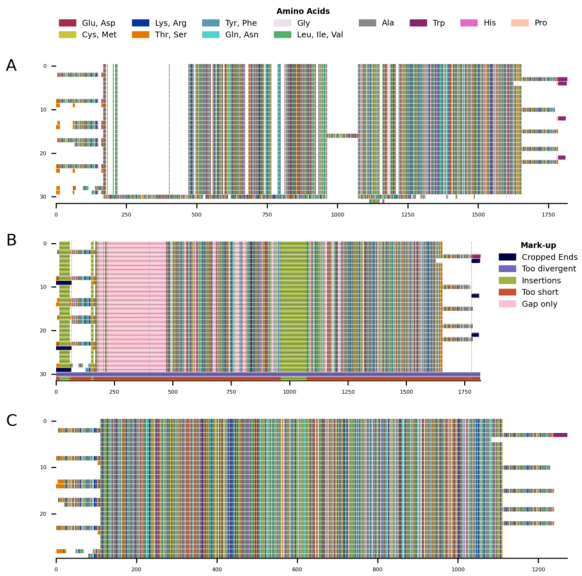
CHAPTER

ONE

CIALIGN

CIAlign is a command line tool which performs various functions to clean, visualise and analyse a multiple sequence alignment (MSA).

1. *Summary*
2. *Citation*
3. *Mailing List*
4. Installation
5. Usage



Example

1.1 Summary

CIAlign allows the user to:

Clean

- Remove sources of noise from an MSA
  - Remove sequences above a threshold level percentage of divergence from the majority.
  - Remove insertions which are not present in the majority of sequences.

  - Crop poorly aligned sequence ends.
  - Remove short sequences below a threshold number of bases or amino acids.
  - Remove columns containing only gaps.
  - Remove either end of an alignment where columns don't meet a minimum identity threshold and coverage level.

Visualise

- Visualise alignments.
  - Generate image files summarising the alignment.
  - Label these images to show how CIAlign has affected the alignment.
  - Draw sequence logos
  - Plot alignment statistics - visualise coverage and conservation at each position in the alignment.

Interpret

- Generate consensus sequences.
- Generate position frequency, position probability and position weight matrices
- Format these matrices to be used as input for the BLAMM and MEME motif analysis tools.
- Generate a similarity matrix showing the percentage identity between each sequence pair.

Edit

- Extract a section of the alignment.
- Unalign the alignment.
- Replace U with T, or T with U in a nucleotide alignment.

CIAlign is designed to be highly customisable, allowing users to specify exactly which functions to run and which settings to use.

It is also transparent, generating a clear log file and alignment markup showing exactly how the alignment has changed and what has been removed by which function.

1.2 Citation

If you found CIAlign useful, please cite:

Tumescheit C, Firth AE, Brown K. 2022. CIAlign: A highly customisable command line tool to clean, interpret and visualise multiple sequence alignments. PeerJ 10:e12983 https://doi.org/10.7717/peerj.12983

1.3 Mailing List

Sign up here for updates when a new feature is added to CIAlign

CHAPTER

TWO

INSTALLATION

Requirements

- python >= 3.6
- matplotlib >= 2.1.1
- numpy >= 1.16.3
- scipy >= 1.3.0

The easiest way to install CIAlign is using conda or pip3.

Conda

```
conda install -c bioconda cialign
```

link

pip3 `pip3 install cialign`

link

Download The current release of CIAlign can also be downloaded directly using this link,

If you download the package directly, you will also need to add the CIAlign directory to your PATH environment variable as described here

CHAPTER

THREE

USAGE

CIAlign is used to process multiple sequence alignments (MSAs) - sets of nucleotide or amino acid sequences which have already been aligned with an external tool.

3.1 Input Files

The input for CIAlign is an aligned MSA in FASTA format.

3.2 Quick Start

```
CIAlign --infile INFILE --outfile_stem STEM OPTIONS
```

Where INFILE is the FASTA file you would like to process, STEM is a prefix for the output files and OPTIONS lists the functions you would like to run and the parameters you would like to use.

For example, to run the remove insertions function, with default settings, on the file `example1.fasta` and generate `ri_clean.fasta`.

```
CIAlign --infile example1.fasta --outfile_stem ri --remove_insertions
```

To run all functions with the default settings (please use this option cautiously):

```
CIAlign --infile example1.fasta --all
```

3.3 Specifying Options

Parameters can be specified in the command line OPTIONS or in a config file.

A template config file is provided in `CIAlign/templates/ini_template.ini` - edit this file and provide the path to the `--inifile` argument.

```
CIAlign --infile INFILE --outfile_stem STEM --inifile my_inifile.ini
```

If this argument is not provided command line arguments and defaults will be used.

Parameters passed in the command line will take precedence over config file parameters, which take precedence over defaults.

Command help can be accessed by typing `CIAlign --help`

3.4 Basic Parameters

Beside these main parameters, the use of every function and corresponding thresholds can be specified by the user by adding parameters to the command line or by setting them in the configuration file. Available functions and their parameters are specified below.

CIAlign always produces a log file, specifying which functions have been run with witch parameters and what has been removed. It also outputs a machine parsable file that only specifies what has been removed with the original column positions and the sequence names.

Output files:

- **`OUTFILE_STEM_log.txt`** - general log file
- **`OUTFILE_STEM_removed.txt`** - removed columns positions and sequence names text file

3.5 Cleaning Functions

The CIAlign cleaning functions are designed to address several common issues with multiple sequence alignments, affecting the speed, complexity and reliability of specific downstream analyses.

All of these functions remove columns or rows from the alignment to address sources of noise.

- *Remove divergent*
- *Remove insertions*
- *Crop ends*
- *Remove short*
- *Remove gap only*
- *Crop divergent*

Each of these steps (if specified) will be performed sequentially in the order specified in the table below.

`remove_divergent`, `remove_insertions`, `crop_ends` and `crop divergent` require three or more sequences in the alignment, `remove_short` and `remove_gap_only` require two or more sequences.

Output files:

The "cleaned" alignment after all steps have been performed will be saved as **`OUTFILE_STEM_cleaned.fasta`**

The *retain* functions allow the user to specify sequences to keep regardless of the CIAlign results.

3.5.1 Remove Divergent

Removes divergent (outlier) sequences from the alignment. It is very common for an MSA to include one or a few outlier sequences which do not align well with the majority of the alignment. For some applications it is useful to remove these.

The remove divergent function specifically removes sequences with <= `remove_divergent_minperc` positions at which the most common residue in the alignment is present.

3.5.2 Remove Insertions

Removes insertions which are not present in the majority of sequences (or regions which are deleted in the majority of sequences). Insertions or other stretches of sequence which are only present in a minority of sequences can lead to large gaps, these are sometimes of interest but can also complicate downstream analysis.

The remove insertions function removes regions from the alignment which are found in <= `insertion_min_perc` of the sequences but are surrounded by >= `insertion_min_flank` columns of higher coverage.

3.5.3 Crop Ends

It is common for an MSA to contain more gaps towards either end than in the body of the alignment, due to (for example) increased sequencing error towards the ends of reads, lower read coverage or assembly issues.

The crop ends function crops the ends of individual sequences if they contain a high proportion of gaps relative to the rest of the alignment. The number of gap positions separating every two consecutive non-gap positions at either end of the sequence is compared to a threshold (calculated from the total sequence length using `crop_ends_mingap_perc`) and if that difference is higher than the threshold, the start of the sequence will be reset to that position.

Note: if the sequences are short (e.g. < 100), a low crop_ends_mingap_perc (e.g. 0.01) will result in a change of gap numbers that is too low (e.g. 0). If this happens, the change in gap numbers will be set to 2 and a warning will be printed.

3.5.4 Remove Short

Removes short sequences below a threshold length.

3.5.5 Remove Gap Only

Removes columns containing only gaps. This function is run by default, to not run this function specify `--keep_gaponly`.

3.5.6 Crop Divergent

Some alignments have a region which is clearly of higher quality than the surrounding alignment, with less diversity and fewer gaps. This can be the case when regions have been extracted, e.g. from a full genome, but the start and end positions of the region of interest are not well defined·

The crop divergent function redefines the start and end positions of an alignment by looking for `crop_divergent_buffer_size` consecutive columns which have a minimum proportion of identical residues >= `crop_divergent_min_prop_ident` and a minimum proportion of non-gap residues >= `crop_divergent_min_prop_nongap`, then taking the first or last such column as the alignment start or end.

3.5.7 Retain

These parameters allow the user to specify sequences which should not be removed from the alignment.

The sequences can be specified by providing one or more sequence names (`--retain`), a character string to match in the names (`--retain_str`) or a file containing a list of names (`--retain_list`).

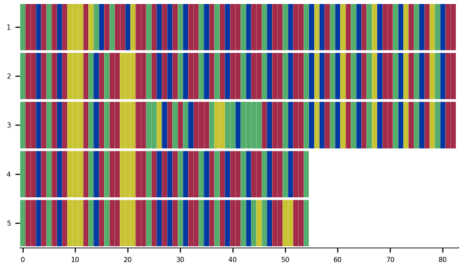
The crop ends, remove divergent and remove short functions also have the option to specify sequence names to ignore with those specific functions only.

3.6 Visualisation functions

Each of these functions produces some kind of visualisation of an MSA.

3.6.1 Mini Alignments

These functions produce "mini alignments" - images showing a small representation of your whole alignment, so that gaps and poorly aligned regions are clearly visible.



Mini alignment example

Output files:

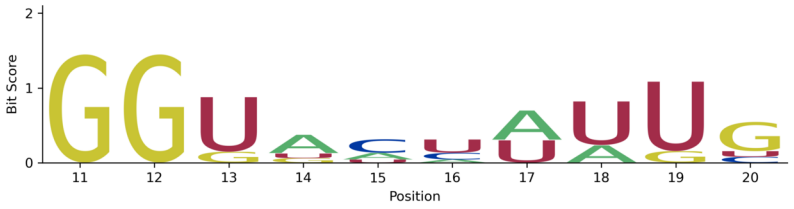
- **`OUTFILE_STEM_input.png (or svg, tiff, jpg)`** - visualisation of the input alignment
- **`OUTFILE_STEM_output.png (or svg, tiff, jpg)`** - visualisation of the cleaned output alignment
- **`OUTFILE_STEM_markup.png (or svg, tiff, jpg)`** - visualisation of the input alignment with deleted rows and columns marked

3.6.2 Sequence logos

These functions draw sequence logos representing output (cleaned) alignment using the algorithm specified by [Schneider (1990)(]https://www.ncbi.nlm.nih.gov/pmc/articles/PMC332411/).

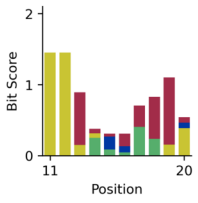
Traditional "text" sequence logos can be produced as well as bar charts summarising the same information.

Text



Text logo example

Bar



Bar logo example

You can also specify a subsection of the alignment using the `logo_start` and `logo_end` arguments, positions should be relative to the input alignment. If no cleaning functions are specified, the logo will be based on your input alignment.

Output_files:

- **`OUTFILE_STEM_logo_bar.png (or svg, tiff, jpg)`** - the alignment represented as a bar chart
- **`OUTFILE_STEM_logo_text.png (or svg, tiff, jpg)`** - the alignment represented as a standard sequence logo using text

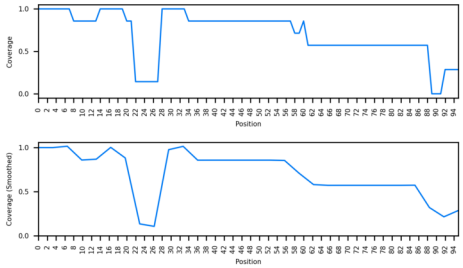
NB: to see available fonts on your system, run CIAlign –list_fonts_only and view CIAlign_fonts.png

3.6.3 Statistics Plots

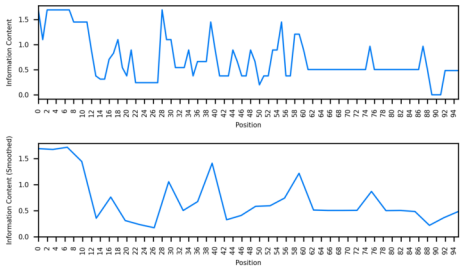
For each position in the alignment, these functions plot:

- Coverage (the number of non-gap residues)
- Information content
- Shannon entropy

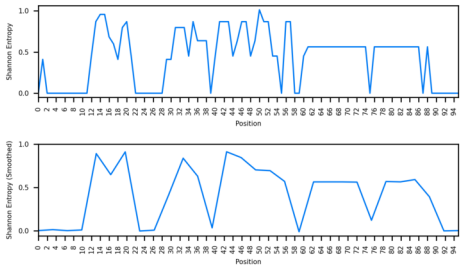
Coverage



Information Content



Shannon Entropy



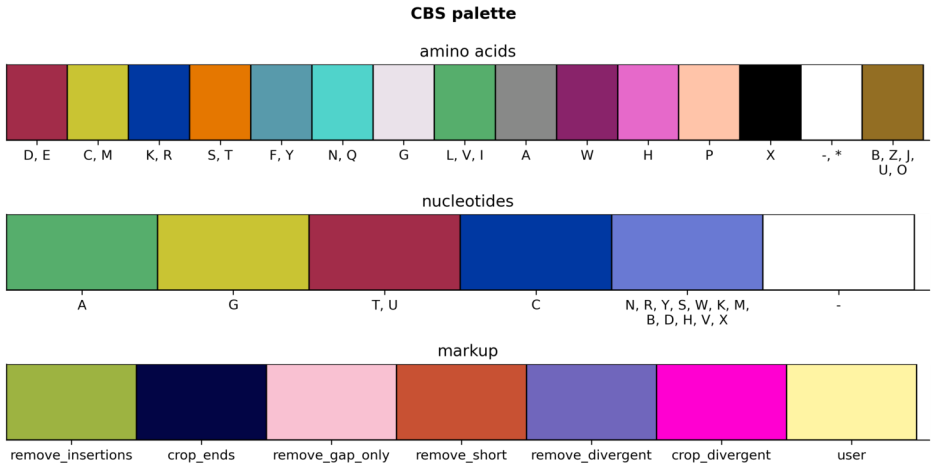
Output files:

- **OUTFILE_STEM_input_coverage.png (or svg, tiff, jpg)** - image showing the input alignment coverage
- **OUTFILE_STEM_output_coverage.png (or svg, tiff, jpg)** - image showing the output alignment coverage
- **OUTFILE_STEM_input_information_content.png (or svg, tiff, jpg)** - image showing the input alignment information content
- **OUTFILE_STEM_output_information_content.png (or svg, tiff, jpg)** - image showing the output alignment information content
- **OUTFILE_STEM_input_shannon_entropy.png (or svg, tiff, jpg)** - image showing the input alignment Shannon entropy
- **OUTFILE_STEM_output_shannon_entropy.png (or svg, tiff, jpg)** - image showing the output alignment Shannon entropy

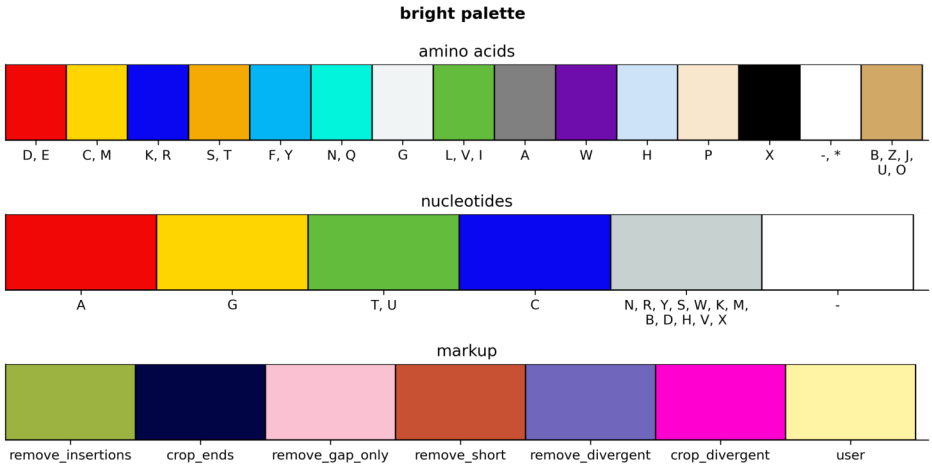
3.6.4 Palettes

This function sets the colour palette for the mini alignments. Currently available palettes are colour blind safe (CBS) and bright.

CBS



CBS palette



Bright Bright palette

3.7 Interpretation Functions

These functions provide additional analyses you may wish to perform on your alignment.

3.7.1 Consensus Sequences

This step generates a consensus sequence based on the cleaned alignment. If no cleaning functions are performed, the consensus will be based on the input alignment.

Consensus sequences can be `majority` - the most common character in each column is used, including gaps or `majority_nongap` - the most common non-gap character is used.

Where the two most frequent characters are equally common a random character is selected.

Once the consensus has been generated, gap positions are automatically removed, specifying `---consensus_keep_gaps` prevents this.

Output files:

- **`OUTFILE_STEM_consensus.fasta`** - the consensus sequence only
- **`OUTFILE_STEM_with_consensus.fasta`** - the cleaned alignment plus the consensus

3.7.2 Position Frequency, Probability and Weight Matrices

These functions are used to create a position weight matrix, position frequency matrix or position probability matrix for your input or output (cleaned) alignment. These are numerical representations of the alignment which can be used as input for various other software, for example to find regions of another sequence resembling part of your alignment. PFMs, PPMs and PWMs are described well in the Wikipedia article here.

You can also specify a subsection of the alignment using the `pwm_start` and `pwm_end` arguments, positions should be relative to the input alignment.

Output_files:

- **`OUTFILE_STEM_pwm_(input/output).txt`** - position weight matrix representing the alignment (or part of the alignment)
- **`OUTFILE_STEM_ppm_(input/output).txt`** - position probability matrix representing the alignment (or part of the alignment)

- **OUTFILE_STEM_pfm_(input/output).txt** - position frequency matrix representing the alignment (or part of the alignment)
- **OUTFILE_STEM_ppm_meme_(input/output).txt** - position probability matrix representing the alignment (or part of the alignment) in the format used by the MEME software suite.
- **OUTFILE_STEM_blamm_(input/output).png** - position probability matrix representing the alignment (or part of the alignment) in the format used by the BLAMM software tool.

3.7.3 Similarity Matrices

Generates a matrix showing the proportion of identical bases / amino acids between each pair of sequences in the MSA.

Output file:

- **OUTFILE_STEM_input_similarity.tsv** - similarity matrix for the input file
- **OUTFILE_STEM_output_similarity.tsv** - similarity matrix for the output file

3.8 Editing Functions

3.8.1 Extracting part of the alignment

This function allows the user to specify a start and end position to isolate part of the alignment, using the –section_start and –section_end position. The section must be at least 5 residues in length. The section which has been isolated will then be used for all other processing with CIAlign.

If parsing functions are also specified, the positions output in the log files will be relative to the original input file, rather than the section.

3.8.2 Replacing U or T

This function replaces the U nucleotides with T nucleotides or vice versa without otherwise changing the alignment.

Output files:

- **OUTFILE_STEM_T_input.fasta** - input alignment with T's instead of U's
- **OUTFILE_STEM_T_output.fasta** - output alignment with T's instead of U's

or

- **OUTFILE_STEM_U_input.fasta** - input alignment with U's instead of T's
- **OUTFILE_STEM_U_output.fasta** - output alignment with U's instead of T's

3.8.3 Unaligning (removing gaps)

This function simply removes the gaps from the input or output alignment and creates and unaligned file of the sequences.

Output files:

- `OUTFILE_STEM_unaligned_input.fasta` - unaligned sequences of input alignment
- `OUTFILE_STEM_unaligned_output.fasta` - unaligned sequences of output alignment