`masking_package` v1.1

Federico Plazzi

July 14, 2016

Contents

1 License **2**

2 Introduction **3**
- 2.1 Background: Alignment Uncertainty . . . 3
- 2.2 What does `masking_package` do? . . . 3
- 2.3 Citing `masking_package` . . . 4
- 2.4 Who wrote `masking_package` . . . 4

3 Running `masking_package` **5**
- 3.1 Requirements . . . 5
- 3.2 Installation and input files . . . 5
- 3.3 Customize sub-scripts . . . 6
- 3.4 Running the package: `masking.sh` and `masking.cfg` files . . . 6
  - 3.4.1 `###CHARSETS###` . . . 7
  - 3.4.2 `###AGREEMENT###` . . . 7
  - 3.4.3 `###DATATYPE###` . . . 7
  - 3.4.4 `###LINELENGTH###` . . . 7
  - 3.4.5 `###SUFFIX###` . . . 7
  - 3.4.6 `###ZORROCUTOFF###` . . . 7
- 3.5 Output files . . . 8
  - 3.5.1 Single masked charsets . . . 8
  - 3.5.2 Final dataset . . . 8
  - 3.5.3 Boundary file . . . 8
  - 3.5.4 Charset summary . . . 8
  - 3.5.5 Keeping details . . . 8
  - 3.5.6 Final plot . . . 8
  - 3.5.7 Masking idiosyncrasies . . . 9

Chapter 1

License

Chapter 2

Introduction

2.1 Background: Alignment Uncertainty

Multiple sequence alignment is a key step in a wide range of biological studies. Many different approaches and softwares have been developed to this purpose; however, despite the striking improvement in alignment accuracy in recent years, regions of uncertainty may still hamper a correct retrieval of positional homologies.
Several metrics were suggested to estimate overall alignment quality (see, e.g., [1]; [2]; [3]; [4]). Measuring site-wise alignment accuracy (i.e. at each individual position), however, is often mandatory, especially in a phylogenetic context, where poorly aligned sites may contribute noise to the tree inference. Those poorly aligned sites should be detected and discarded (see, e.g., [5]; [6]; [7]): this task is typically known as trimming or masking.
Masking can be carried out through many different methods and tools: the present package, which is called "`masking_package`", was written to call five different masking tools, compare their results and produce a final consensus accounting for all of them.

2.2 What does `masking_package` do?

`masking_package` was written in `bash` and R languages to perform and compare masking from different softwares.
It is also possible to provide separate alignments (e.g., separate genes) in the same analysis: they will be separately aligned, masked, and evaluated; then, they will be concatenated in the final resulting dataset.

1. At first, `masking_package` performs multiple sequence alignment (MSA) using T-Coffee [8].
   T-Coffee was chosen because of its large versatility: it implements many different alignment methods and algorithms and it can even combine more of them in a single resulting alignment (the M-Coffee mode; [9]).

2. Five different masking strategies are applied to T-Coffee MSAs. Five different tools are called to this aim; namely,

   Aliscore [10];

   BMGE [11];

   GBlocks [6];

   Noisy [12];

   Zorro [13].

3. Outputs from different masking strategies are compared and final MSAs are produced keeping only sites selected as phylogenetically informative by at least k tools, where k is up to the user.

4. If more than one MSA is provided, all the resulting MSAs are concatenated in a final dataset.

2.3 Citing `masking_package`

If you include `masking_package` in your publication, please cite [14]:

Plazzi F, Puccio G, Passamonti M. Comparative large-scale mitogenomics evidence clade-specific evolutionary trends in mitochondrial DNAs of Bivalvia. Genome Biol Evol. Forthcoming.

All the tools called by `masking_package` should be cited as well; eventually, R [15] and the package `seqinr` [16] should also be given proper credits.

2.4 Who wrote `masking_package`

`masking_package` was written by Federico Plazzi at the Department of Biological, Geological and Environmental Sciences of the University of Bologna. Please report any bug to

`federico [dot] plazzi [at] unibo [dot] it`

Chapter 3

Running masking_package

3.1 Requirements

masking_package is a suite of different scripts. Most of them, including the main one (masking.sh) are bash scripts, and two R scripts are called as well. To download and install the R environment, follow instructions at its website.

http://www.r-project.org/

The additional package seqinr is loaded by R in order to perform most analyses. To install it, just type

```
> install.packages("seqinr")
```

at the R prompt and follow on-screen instructions. You may need root privileges.
Obviously, you also need T-Coffee, Aliscore, BMGE, GBlocks, Noisy and Zorro to be properly installed on your machine. Specific commands for these six softwares are open to the user for modifications (see 3.3), therefore you do not need specific paths nor locations.

3.2 Installation and input files

Once the archive is extracted in the masking_package folder, the package is ready to be set up. Alignment files must be in the parent directory. They must be in the FASTA format.

3.3 Customize sub-scripts

Provided sub-scripts for T-Coffee (`t_coffee_cmd.sh`), Aliscore (`aliscore_cmd.sh`), BMGE (`BMGE_cmd.sh`), GBlocks (`GBlocks_cmd.sh`), Noisy (`noisy_cmd.sh`) and Zorro (`zorro_cmd.sh`) must be carefully assessed by the user and are not provided ready-to-use. There are mainly two reasons for this.

1. Specific installations need different syntaxes to call these tools (e.g., bin names, paths, ...)
2. Options for each of these tools need to be fine-tuned on the specific features of the study (e.g., matrices, algorithms, cutoff values, ...)

There are two main constraints that must be followed while customizing sub-scripts.

1. Input FASTA files must have the extension `.fas`. All the files with this extension will be copied to the T-Coffee folder: however, it is possible to have T-Coffee align only some of them by specifying, e.g., "`for i in *_aa.fas`" or "`for i in *_prot.fas`" in the `t_coffee_cmd.sh` script.
2. The MSAs resulting from T-Coffee must have the extension `.fasta_aln`, because this is the extension of the files that will be copied to the folders of masking tools. This can be easily achieved by specifying "`-output=fasta_aln`" in the `t_coffee_cmd.sh` script (see the provided script).

When more alignments are requested, it is useful to use a `for` statement, as in the provided scripts; for example

```
for i in *.fasta_aln
do perl -I /opt/Aliscore_v.2.0 /opt/Aliscore_v.2.0/Aliscore.02.2.pl
-N -r 6555 -w 30 -i "$i"
done
```

(from the `aliscore_cmd.sh` script).

3.4 Running the package: `masking.sh` and `masking.cfg` files

The main script is `masking.sh`. It calls all the sub-scripts (see 3.3), as well as the two R scripts. All the analysis can be started by simply typing

```
> sh masking.sh
```

at the command line from within the `masking_package` directory. Depending on your installation, you may need root privileges, e.g. for running T-Coffee. However, before running it, you may consider to comment out line 10 if you do

not want to eliminate dashes and to change "X" into "N" at line 13 if you are aligning nucleotides instead of amino acids.
Moreover, you have to set up the `masking.cfg` file that will be used by R to configure most parameters.
All the parameters are listed immediately after the proper between-octothorpes header (i.e., no blank lines are allowed between the header and the parameter value); there is no specific order for options. Below all options are briefly explained and the default value is shown between brackets.

3.4.1 ###CHARSETS###

[*no default*] These are alignment names ("charsets"), that typically correspond to genes; at least one charset must be listed. These charsets will be used to build the names of files resulting from T-Coffee and masking tools. It is possible to specify a suffix to be applied immediately after the charset name using the option `###SUFFIX###` (see 3.4.5); the standard extensions `.fas` and `.fasta_aln` will be applied, as well as other tool-specific extensions.

3.4.2 ###AGREEMENT###

[3] This option sets k, the number of softwares that must have retained a site to be retained also in the final alignment produced by `masking_package`. Obviously, $1 \leq k \leq 5$; otherwise, the script stops with an error.

3.4.3 ###DATATYPE###

[AA] Only one datatype is allowed: set it to "AA" for amino acids and "DNA" for nucleotides.

3.4.4 ###LINELENGTH###

[60] Length of each line in the FASTA outputs.

3.4.5 ###SUFFIX###

[*blank*] A suffix that is possible to have append to charset names before adding proper extensions. A blank line after the header is allowed and specifies a blank string (which is the default).

3.4.6 ###ZORROCUTOFF###

[4] Zorro cutoff value: sites with a score equal to this or above will be retained in the Zorro analysis. This value ranges from 0 to 10 and must be divided by 10 to obtain the probability value which is described in the original publication [13].

3.5 Output files

All the output files, if not otherwise stated, are produced in the parent directory.

3.5.1 Single masked charsets

FASTA alignment of single masked charsets are produced in the `Masking` subfolder which is created in the parent directory.

3.5.2 Final dataset

The final dataset is obtained by concatenating all the single masked charsets (see 3.5.1). Whenever a taxon is missing from a given MSA, a stretch of gaps ("`-`") is used in the final dataset alignment in that region.
This file is called "`dataset_masked.fas`" and is in FASTA format.

3.5.3 Boundary file

A boundary file (`boundaries.cfg`) is produced reporting start and end point of each charset in the final concatenation. It is formatted for direct use into PartitionFinder [17].

3.5.4 Charset summary

The number of taxa detected in each MSA is detailed in the charset summary (`charset_nums.txt`).

3.5.5 Keeping details

`kept.msk`, `masking.msk` and `masking_agreement.msk` are internal interchange files which list all the sites that were retained for each charset and site-wise results for each tool.

3.5.6 Final plot

The final plot is printed as `masking.pdf`. For each charset and for all the possible values of k (i.e., $k \in \{1, 2, 3, 4, 5\}$), this plot shows three different statistics.

1. The total amount of selected sites.
2. The percentage of the total amount of selected sites over the original number of sites, N_o, defined as

$$N_o = \sum_{i=1}^{S} l_i \tag{3.1}$$

   where l_i is the length of the i-th sequence and S is the total number of sequences.

3. the percentage of the total amount of selected sites over the aligned number of sites, N_a, defined as

$$N_a = S \cdot l_a \tag{3.2}$$

where S is the total number of sequences and l_a is the length of the MSA.

3.5.7 Masking idiosyncrasies

Different masking strategies may behave differently for the same site. To grossly check for consistency between masking tools, the `masking_agreement.pdf` plot is printed. The first column refers to the total length of the final (concatenated) MSA; all the other columns refer to the number of sistes selected by all the possible combinations of softwares. Ideally, only the `noone` (leftmost) and the `all` (rightmost) columns should be significant, meaning that all softwares were generally in agreement in discarding or keeping, respectively.

Bibliography

[1] Schwartz AS, Pachter L. Multiple alignment by sequence annealing. Bioinformatics. 2007;23: e24–29.

[2] Rosenberg MS. Evolutionary distance estimation and fidelity of pair wise sequence alignment. BMC Bioinformatics. 2005;6: 102.

[3] Lassmann T, Sonnhammer EL. Quality assessment of multiple alignment programs. FEBS Lett. 2002;529: 126-130.

[4] Thompson JD, Plewniak F, Poch O. A comprehensive comparison of multiple sequence alignment programs. Nucleic Acids Res. 1999;27: 2682-2690.

[5] Loytynoja A, Goldman N. Phylogeny-aware gap placement prevents errors in sequence alignment and evolutionary analysis. Science. 2008;320: 1632-1635.

[6] Castresana J. Selection of conserved blocks from multiple alignments for their use in phylogenetic analysis. Mol Biol Evol. 2000;17: 540-552.

[7] Grundy WN, Naylor GJ. Phylogenetic inference from conserved sites alignments. J Exp Zool. 1999;285: 128-139.

[8] Notredame C, Higgins DG, Heringa J. T-Coffee: A Novel Method for Fast and Accurate Multiple Sequence Alignment. J Mol Biol. 2000;302: 205-217.

[9] Wallace IM, O'Sullivan O, Higgins DG, Notredame C. M-Coffee: combining multiple sequence alignment methods with T-Coffee. Nucleic Acids Res. 2006;34: 1692-1699.

[10] Misof B, Misof K. A Monte Carlo approach successfully identifies randomness in multiple sequence alignments: a more objective means of data exclusion. Syst Biol. 2009;58: 21-34.

[11] Criscuolo A, Gribaldo S. BMGE (Block Mapping and Gathering with Entropy): selection of phylogenetic informative regions from multiple sequence alignments. BMC Evol Biol. 2010;10: 210.

[12] Dress AWM, Flamm C, Fritzsch G, Grünewald S, Kruspe M, Prohaska SJ, Stadler PF. Noisy: Identification of problematic columns in multiple sequence alignments. Algorithm Mol Biol. 2008;3: 7.

[13] Wu M, Chatterji S, Eisen JA. Accounting For Alignment Uncertainty in Phylogenomics. PLoS One. 2012;7: e30288.

[14] Plazzi F, Puccio G, Passamonti M. Comparative large-scale mitogenomics evidence clade-specific evolutionary trends in mitochondrial DNAs of Bivalvia. Genome Biol Evol. Forthcoming.

[15] R Development Core Team. R: A language and environment for statistical computing. 2008; Vienna: R Foundation for Statistical Computing.

[16] Charif D, Lobry JR. SeqinR 1.0-2: a contributed package to the R project for statistical computing devoted to biological sequences retrieval and analysis. In Bastolla U, Porto M, Roman HE, Vendruscolo M (eds.). Structural approaches to sequence evolution: Molecules, networks, populations. 2007; New York: Springer Verlag. p. 207-232.

[17] Lanfear R, Calcott B, Ho SYW, Guindon S. PartitionFinder: combined selection of partitioning schemes and substitution models for phylogenetic analyses. Mol Biol Evol. 2012;29: 1695-1701.