BAli-Phy Tutorial (for version 3.4)

Benjamin Redelings

Table of Contents

1. Introduction

This tutorial offers a quick overview of bali-phy with an emphasis on concrete analysis of real datasets. Other documentation includes:

- The Users Guide is much more complete.
- Run ***tool*** **--help** to see command line options for *tool*.
- Run **man** ***tool*** to see the manual page for *tool*.

- Manual pages for bali-phy and tools are also available online.

Before you start this tutorial, please download and install bali-phy, following the installation instructions in the User Guide.

2. Setting up the `~/alignment_files` directory

Go to your home directory:

```
% cd ~
```

Make a directory called alignment_files inside it:

```
% mkdir alignment_files
```

Go into the `alignment_files` directory:

```
% cd alignment_files
```

Download the example alignment files:

```
% wget http://www.bali-phy.org/examples.tgz
```

Alternatively, you can use **curl**

```
% curl -O http://www.bali-phy.org/examples.tgz
```

Extract the compressed archive:

```
% tar -zxf examples.tgz
```

Take a look inside the `examples` directory:

```
% ls examples
```

Take a look at an input file (you can press 'q' to exit 'less'):

```
% less examples/5S-rRNA/5d.fasta
```

Get some information about the alignment:

```
% alignment-info examples/5S-rRNA/5d.fasta
```

3. Command line options

What version of bali-phy are you running? When was it compiled? Which compiler? For what computer type?

```
% bali-phy -v
% bali-phy --version
```

Look at the list of command line options:

```
% bali-phy -h     | less
% bali-phy --help | less
% bali-phy help   | less
```

By default, not all the options are shown. To see more options, try:

```
% bali-phy help advanced | less
% bali-phy help expert   | less
```

All bali-phy tools also take `--help`.

```
% alignment-thin --help | less
% alignment-cat --help  | less
```

4. Help

You can get extended help on each command line option:

```
% bali-phy help iterations | less        // --iterations=<num> command line option
% bali-phy help alphabet   | less        // --alphabet=<a> command line option
```

You can also get help on models, distributions, functions, and command-line options:

```
% bali-phy help tn93       | less        // Tamura-Nei (1993) model
% bali-phy help normal     | less        // Normal distribution
% bali-phy help quantile   | less        // quantile function
```

There are additional help topics you might want to explore:

```
% bali-phy help topics     | less        // to see additional help topics
% bali-phy help parameters | less        // setting parameters
% bali-phy help Codons     | less        // Codons alphabets
```

5. Analysis 1: 5S rRNA -- free alignment versus fixed alignment

Let's estimate a phylogeny on a very small data set under both a free alignment and a fixed alignment to see how the alignment affects our confidence in the estimated topology.

5.1. Free alignment

Let's get started by going to the examples directory:

```
% cd ~/alignment_files/examples/5S-rRNA
```

Now lets start a run. BAli-Phy will create a directory called `5d-free-1` to hold output files from this analysis, as directed by the **`-n 5d-free`** option. Here **`-n`** is short for **`--name`**, while **`-S`** is a short form of **`--smodel`** and specifies the substitution model.

```
% bali-phy 5d-clustalw.fasta -S gtr+Rates.gamma[4]+inv -n 5d-free &
```

Yay! Now you have started your first copy of the anlysis. Let's run another copy of the same analysis, which will create another directory called `5d-free-2` for its own output files. This additional run will take advantage of a second processor, and will also help detect with the runs have performed enough iterations.

```
% bali-phy 5d-clustalw.fasta -S gtr+Rates.gamma[4]+inv -n 5d-free &
```

You can take a look at jobs running in the background. There should be two bali-phy jobs running:

```
% jobs
```

The previous command only shows jobs started by the shell you are typing in. You can also examine all processes running on your computer. bali-phy should show up near the top since it using a lot of the CPU:

```
% top          // press 'q' to exit top
```

In order to see how many iterations have completed, check the number of lines in the log files:

```
% wc -l 5d-*/C1.log
```

To summarize the results generated so far, run the summary script and then view the results in a web browser:

```
% bp-analyze 5d-free-1/ 5d-free-2/
% firefox Results/index.html &
```

5.1.1. Alignment questions:

- How much difference is there between the estimated alignment and the initial alignment? (See *Alignment Distribution / Initial / Diff*)
- Which part of the alignment is most certain? (See *Alignment Distribution / Best (WPD) / AU*)

5.1.2. Parameter questions:

- What is the posterior median for **inv:p_inv**? (See *Scalar Variables*)
- What is the meaning of **inv:p_inv** parameter? (Run **bali-phy help inv**)
- What is the prior distribution on **inv:p_inv**? (See *Model and priors / Substitution model*)

When you have about 2000 samples from each run, summarize the latest results:

```
% wc -l 5d-*/C1.log
% bp-analyze 5d-free-1/ 5d-free-2/
% mv Results 5d-free.Results
% firefox 5d-free.Results/index.html &
```

Now lets kill the bali-phy processes:

```
% top         // use top to find the PID (process id) for each of the bali-phy processes
% kill pid1
% kill pid2
% top         // check that the jobs have stopped.
```

Now let's look at the tree in figtree:

```
% figtree 5d-free.Results/c50.PP.tree &
```

1. When it says "Please select a name for these values", enter **PP**.
2. Then, enable "Branch Labels".
3. Under "Branch Labels / Display", select "PP".
4. You can increase the font size to make the figure more readable.
5. You can also increase the font size of the tip labels.

5.1.3. Tree questions (free alignment):

- How many internal branches are in the 50% consensus tree?
- What is the posterior probability for each of these internal branches?

5.2. Fixed alignment

```
% cd ~/alignment_files/examples
% bali-phy 5d-clustalw.fasta -S gtr+Rates.gamma[4]+inv -I none -n 5d-fixed &
% bali-phy 5d-clustalw.fasta -S gtr+Rates.gamma[4]+inv -I none -n 5d-fixed &
```

The **-I none** is a short form of **--imodel=none**, where *imodel* means the insertion-deletion model. When there's no model of insertions and deletions, then the alignment must be kept fixed.

Summarize the analysis after about 2000 iterations:

```
% wc -l 5d-*/C1.log
% bp-analyze 5d-fixed-1/ 5d-fixed-2
```

Examine the majority consensus tree for the fixed alignment analysis:

```
% figtree Results/c50.PP.tree &
```

5.2.1. Tree questions (fixed alignment):

- How many internal branches are in the 50% consensus tree?
- What is the posterior probability for each of these internal branches?
- For this data set, how does fixing the alignment affect our confidence in the tree? Why might this be?

You can kill the remaining runs after you answer these questions:

```
% killall bali-phy
```

However, beware: if you are running multiple analyses, this will terminate all of them.

6. Analysis 2: ITS sequences -- multi-gene analysis

Let's do an analysis of intergenic transcribed spacer (ITS) genes from 20 specimens of lichenized fungi (Gaya et al, 2011 analyzes 68 specimens). The ITS genes contain a lot of variation, but are hard to align. This analysis will show how to estimate the alignment, phylogeny, and evolutionary parameters using MCMC. Averaging over all alignments makes it safe to use the ITS region without throwing out ambiguously-aligned regions.

This data set is divided into three gene regions, or partitions. It is assumed that all genes evolve on the same tree, but may have different rates and evolutionary parameters. Let's look at the sequences. How long are they?

```
% cd ~/alignment_files/examples/ITS
% alignment-info ITS1.fasta
% alignment-info 5.8S.fasta
% alignment-info ITS2.fasta
```

By default, each gene gets a default substitution model based on whether it contains DNA/RNA or amino acids. By running bali-phy with the `--test` option, we can reveal what substitution models and priors will be used, without actually starting a run.

```
% bali-phy ITS1.fasta 5.8S.fasta ITS2.fasta --test
```

Let's run two copies of this analysis:

```
% bali-phy ITS1.fasta 5.8S.fasta ITS2.fasta &
% bali-phy ITS1.fasta 5.8S.fasta ITS2.fasta &
```

Since we did not specify a name for the analysis, bali-phy creates the name `ITS1-5.8S-ITS2` from the input file names. The mean or the median of these values can be used as an estimate of the parameter. The variance indicates the degree of uncertainty. Let's look at the initial parameter estimates:

```
% statreport ITS1-5.8S-ITS2-1/C1.log ITS1-5.8S-ITS2-2/C1.log | less
% statreport ITS1-5.8S-ITS2-1/C1.log ITS1-5.8S-ITS2-2/C1.log --mean | less
```

The parameter estimates are also summarized in a table in the summary report:

```
% bp-analyze ITS1-5.8S-ITS2-1/ ITS1-5.8S-ITS2-2/
% firefox Results/index.html &
```

The program Tracer graphically displays the posterior probability distribution for each parameter.

```
% tracer ITS1-5.8S-ITS2-1/C1.log ITS1-5.8S-ITS2-2/C1.log &
```

If you are using Windows or Mac, first run Tracer, and then press the + button to add the files.

6.1. Questions about parameter differences between genes

These genes have different evolutionary processes. How does the evolutionary process for these genes differ in:

1. substitution rate? (**`Scale[1]`**, **`Scale[2]`**, ...)
2. insertion-deletion rate? (**`I1/rs07:log_rate`**, **`I2/rs07:log_rate`**, ...)
3. nucleotide frequencies? (**`S1/tn93:pi[A]`**, **`S1/tn93:pi[C]`**, ...)
4. number of indels? (**`#indels`**)

7. Analysis 3: Exons and Introns

Let's look at a data set containing both exons and introns. The pattern of dividing a sequence into high-rate and low-rate regions also applies to stems and loops in RNA sequences. It is helpful to allow different insertion-deletion rates in conserved and non-conserved regions of a gene.

7.1. Extracting parts of a gene

First, lets split the gene into separate FASTA files for each intron and exon:

```
% cd ~/alignment_files/examples/ferns/
% alignment-cat -c 5-8       -e cleaned.fasta > exon1.fasta
% alignment-cat -c 9-453     -e cleaned.fasta > intron1.fasta
% alignment-cat -c 454-596   -e cleaned.fasta > exon2.fasta
% alignment-cat -c 597-864   -e cleaned.fasta > intron2.fasta
% alignment-cat -c 865-948   -e cleaned.fasta > exon3.fasta
% alignment-cat -c 949-1328  -e cleaned.fasta > intron3.fasta
% alignment-cat -c 1329-1393 -e cleaned.fasta > exon4.fasta
```

Now, let's put the pieces back together in case we need the complete gene in the future:

```
% alignment-cat exon1.fasta intron1.fasta exon2.fasta intron2.fasta exon3.fasta intron3.fasta exon4.fasta >
combined.fasta
```

It is also possible to refer to character sets without creating a file for each one, as mentioned below.

7.2. The Rates.free model

We will add `-S gtr+Rates.free[n=4]` to use the free-rates model of substitution with 4 categories:

```
% bali-phy exon1.fasta intron1.fasta exon2.fasta intron2.fasta exon3.fasta intron3.fasta exon4.fasta -S
gtr+Rates.free[n=4] --test
```

This model is more general than Rates.gamma, since it estimates 4 free rates and the frequencies of the 4 rates. These 8 parameters have 6 degrees of freedom, which is a lot more than the 1 degree of freedom for the `Rates.gamma` model. Furthermore, each partition by default gets a separate copy of the model, which leads to 42 degrees of freedom just for the `Rates.free` part of the model!

7.3. Fixing the alignment of some partitions

We will now add `-I 1,3,5,7:none` to disable alignment estimation, but only for the exons:

```
% bali-phy exon1.fasta intron1.fasta exon2.fasta intron2.fasta exon3.fasta intron3.fasta exon4.fasta -S
gtr+Rates.free[n=4] -I 1,3,5,7:none --test
```

7.4. Linking parameters between partitions

We can reduce the number of parameters by using one set of parameters for the exons, and one set for the introns. This is called *linking* the models.

```
% bali-phy exon1.fasta intron1.fasta exon2.fasta intron2.fasta exon3.fasta intron3.fasta exon4.fasta -I 1,3,5,7:none -S
1,3,5,7:gtr+Rates.free[n=4] -S 2,4,6:gtr+Rates.free[n=4] --test
```

We would also like to share the indel model between introns:

```
% bali-phy exon1.fasta intron1.fasta exon2.fasta intron2.fasta exon3.fasta intron3.fasta exon4.fasta -I 1,3,5,7:none -S
1,3,5,7:gtr+Rates.free[n=4] -S 2,4,6:gtr+Rates.free[n=4] -I 2,4,6: --test
```

7.5. Character sets

Instead of creating separate files for each intron and exon, we can also refer to character sets of the combined file:

```
% bali-phy cleaned.fasta:5-8 cleaned.fasta:454-596 cleaned.fasta:865-948 cleaned.fasta:1329-1393 cleaned.fasta:9-453
cleaned.fasta:597-864 cleaned.fasta:949-1328 -I 1,2,3,4:none -S 1,2,3,4:gtr+Rates.free[n=4] -S
5,6,7:gtr+Rates.free[n=4] -I 5,6,7: --test
```

However, the command line is getting so long that it is difficult to manage.

7.6. Option files

Instead of writing all options on the command line, We can put some of them into a text file that is easier to manage and edit. Let's make a text file called `options1.txt`:

```
align = cleaned.fasta:5-8       # exon1
align = cleaned.fasta:9-453     # intron1
align = cleaned.fasta:454-596   # exon2
align = cleaned.fasta:597-864   # intron2
align = cleaned.fasta:865-948   # exon3
align = cleaned.fasta:949-1328  # intron3
align = cleaned.fasta:1329-1393 # exon4

# exons
smodel = 1,3,5,7:gtr+Rates.free[n=4]
imodel = 1,3,5,7:none

# introns
smodel = 2,4,6:gtr+Rates.free[n=4]
imodel = 2,4,6:
```

We can then run

```
% bali-phy -c options1.txt --test
```

Let's run two copies of this analysis:

```
% bali-phy -c options1.txt &
% bali-phy -c options1.txt &
```

7.7. Questions about intron alignments

- How do bali-phy alignments of the introns differ from muscle and mafft alignments of the introns?
- How much certainty or uncertainty do the intron alignments have?
- How do the evolutionary rates of the exons compare to the evolutionary rates of the introns?
- Would it be reasonable to link the intron rates? How about the exon rates?

7.8. Exons as codons + introns as nucleotides

Another thing that we can do is to use a codon model on the combined exons, while using a nucleotide model on the introns. In order to load nucleotide sequences as codons, three conditions must be met:

1. **Sequence lengths must be multiples of three.**

   `AAA --- TTT` is OK

   `AAA --- T--` is not OK.

2. **Gaps must be codon-aligned.**

   `AAA --- TTT` is OK

   `AA- --A TT-` is not OK.

3. **The sequences must be in-frame and not have stop codons.**

   `ATG AAT AAA` is OK, because it translates to `MNK`.

   `AAT GAA TAA` is not OK, because it translates to `NE*`, where`*` is a stop codon.

Note that spaces in FASTA files are allowed, but not required.

Let's first extract the coding data:

```
% alignment-cat exon{1,2,3,4}.fasta -c2-295 > coding.fasta     // Concatenate exons and chop off partial codons at
beginning and end
% sed -i 's/-/N/g' coding.fasta                                // Change codons like A-- to ANN at end of sequence
```

We can then construct an option file `options2.txt`:

```


align = intron1.fasta
align = intron2.fasta
align = intron3.fasta
align = coding.fasta

smodel = 1,2,3:gtr+Rates.free[n=4]
imodel = 1,2,3:rs07
```

```
smodel = 4:gy94
#smodel = 4:gy94_ext[gtr_sym]     # A GTR-based version of GY94 a.k.a. M0.
#smodel = 4:gtr+fMutSel0
#smodel = 4:function[w,fMutSel0[omega=w]]]+m3
imodel = 4:none
```

We can choose any one of the three codon models for the coding sequences in the 4th partition.

```
% bali-phy -c options2.txt --test
```

Try changing the model to **`fMutSel0`**. Does the number of parameters increase or decrease?

7.9. Exons as amino acids + introns as nucleotides

We can also translate the coding sequence into a protein sequence, while using a nucleotide model on the introns. Let's first extract the coding data:

```
% alignment-translate < coding.fasta > protein.fasta
```

We can then construct an option file `options3.txt`:

```
align = intron1.fasta
align = intron2.fasta
align = intron3.fasta
align = protein.fasta

smodel = 1,2,3:gtr+Rates.free[n=4]
imodel = 1,2,3:rs07

smodel = 4:lg08+Rates.log_normal[n=4]
imodel = 4:none
```

```
% bali-phy -c options3.txt --test
```

8. Analysis 4: ITS sequences - with a better model

Let's revisit the ITS sequences used in analysis 2. Now that you've been exposed to different models in different partitions, lets use a more complex substitution model for the ITS partitions (**`--smodel 1,3:tn93+Rates.free[n=3]`**). Note that this also links the substitution models for the ITS partitions.

We can also fix the alignment for 5.8S partition, since it has almost no indels.

```
align = ITS1.fasta
align = 5.8S.fasta
align = ITS2.fasta

smodel = 1,3:tn93+Rates.free[n=3]
smodel = 2:tn93

imodel = 2:none

scale = 1,3:
```

This file additionally links the *scale* for the two ITS partitions (**`--scale=1,3:`**). This forces them to share the same evolutionary rate, and allows more precise estimates of the shared scale. You can run the analysis with unlinked scales by removing this option in order to determine if (i) the scales are similar enough to share and (ii) the scales have a high enough variance that they need to be linked.

9. Complex substitution models

While those analyses are running, let's look at how to specify more complex substitution models in bali-phy.

9.1. Defaults

When you don't specify values for parameters like *imodel*, bali-phy uses sensible defaults. For example, these two commands are equivalent:

```
% cd ~/alignment_files/examples/
% bali-phy 5S-rRNA/25-muscle.fasta --test
% bali-phy 5S-rRNA/25-muscle.fasta --test --alphabet=RNA --smodel=tn93 --imodel=rs07
```

You can change the substitution model from the Tamura-Nei model to the General Time-Reversible model:

```
% bali-phy 5S-rRNA/25-muscle.fasta --test -S gtr
```

Here the `-S gtr` is a short form of `--smodel=gtr`, where *smodel* means the substitution model.

9.2. Rate variation

You can also allow different sites to evolve at 5 different rates using the gamma[4]+INV model of rate heterogeneity:

```
% bali-phy 5S-rRNA/25-muscle.fasta --test -S gtr+Rates.gamma[4]+inv
```

You can also use a log_normal instead of a gamma. log_normal+INV is sometimes better behaved than gamma+INV, because the smallest rate bin under log_normal is not quite as close to 0.

```
% bali-phy 5S-rRNA/25-muscle.fasta --test -S gtr+Rates.log_normal[4]+inv
```

You can allow 5 different rates that are all independently estimated:

```
% bali-phy 5S-rRNA/25-muscle.fasta --test -S gtr+Rates.free[n=5]
```

9.3. Codon models

We can also conduct codon-based analyses using the Nielsen and Yang (1998) model of diversifying positive selection (dN/dS):

```
% bali-phy Globins/bglobin.fasta --test -S gy94[pi=f1x4]
```

The gy94_ext model extends the gy94 model by taking a nucleotide exchange model as a parameter. The gy94 model is equivalent to gy94_ext[hky85_sym]. However, if you want a more flexible codon model, you could use gtr_sym:

```
% bali-phy Globins/bglobin.fasta --test -S gy94_ext[gtr_sym,pi=f1x4]
```

You can make the codon frequencies to be generated from a single set of nucleotide frequencies:

```
% bali-phy Globins/bglobin.fasta --test -S mg94_ext[gtr]
```

The M7 model allows different sites to have different dN/dS values, where the probability of dN/dS values follows a beta distribution:

```
% bali-phy Globins/bglobin.fasta --test -S m7
```

The M7 model has parameters as well. Here are the defaults:

```
% bali-phy Globins/bglobin.fasta --test -S function[w,gy94[omega=w]]+m7
```

The M3 model allows different sites to have different dN/dS values, but directly estimates what these values are:

```
% bali-phy Globins/bglobin.fasta --test -S m3[n=3]
```

The M8a_Test model allows testing for positive selection in some fraction of the sites:

```
% bali-phy Globins/bglobin.fasta --test -S function[w,gy94[omega=w,pi=f3x4]]+m8a_test
```

9.4. Fixing parameter values

We can use the TN93+Gamma[4]+INV model without specifying parameters:

```
% bali-phy Globins/bglobin.fasta --test -S tn93+Rates.gamma+inv
```

However, we can also fix parameter values:

```
% bali-phy Globins/bglobin.fasta --test -S tn93+Rates.gamma[n=4,alpha=1]+inv[p_inv=0.2]
```

Here we have set the shape parameter for the Gamma distribution to 1, and the fraction of invariant sites to 20%. Since these parameters are fixed, they will not be estimated and their values will not be shown in the log file.

You can see the parameters for a model by using the `help` command, as in:

```
% bali-phy help Rates.gamma
```

This will show the default value or default prior for each parameter, if there is one.

9.5. Priors

By default the fraction of invariant sites follows a uniform[0,1] distribution:

```
% bali-phy help inv
```

However, we can specify an alternative prior:

```
% bali-phy Globins/bglobin.fasta --test -S tn93+Rates.gamma[n=4]+inv[p_inv~uniform[0,0.2]]
```

We can also specify parameters as positional arguments instead of using variable names:

```
% bali-phy Globins/bglobin.fasta --test -S tn93+Rates.gamma[4]+inv[~uniform[0,0.2]]
```

Here "`~`" indicates a sample from the uniform distribution instead of the distribution itself.

The insertion-deletion model also has parameters.

```
% bali-phy help rs07
```

Here the default value for rs07:mean_length is exponential[10,1]. This indicates a random value that is obtained by sampling an Exponential random variable with mean 10 and then adding 1 to it.

10. Specifying the model for each partition

For analyses with multiple partitions, we might want to use different models for different partitions. When two partitions have the same model, we might also want them to have the same parameters. This is described in more detail in section 4.3 of the manual.

10.1. Using different substitution models

Now lets specify different substitution models for different partitions.

```
% cd ~/alignment_files/examples/ITS
% bali-phy {ITS1,5.8S,ITS2}.fasta -S 1:gtr -S 2:hky85 -S 3:tn93 --test
```

10.2. Disabling alignment estimation for some partitions

We can also disable alignment estimation for some, but not all, partitions:

```
% bali-phy {ITS1,5.8S,ITS2}.fasta -I 1:rs07 -I 2:none -I 3:rs07 --test
```

Specifying `-I none` removes the insertion-deletion model and parameters for partition 2 and also disables alignment estimation for that partition.

Note that there is no longer an I3 indel model. Partition #3 now has the I2 indel model.

10.3. Sharing model parameters between partitions

We can also specify that some partitions with the same model also share the same parameters for that model:

```
% bali-phy {ITS1,5.8S,ITS2}.fasta -S 1,3:gtr -I 1,3:rs07 -S 2:tn93 -I 2:none --test
```

This means that the information is *pooled* between the partitions to better estimate the shared parameters.

Take a look at the model parameters, and the parentheticals after the model descriptions. You should see that there is no longer an S3 substitution model or an I3 indel model. Instead, partitions #1 and #3 share the S1 substitution model and the I1 indel model.

10.4. Sharing substitution rates between partitions

We can also specify that some partitions share the same scaling factor for branch lengths:

```
% bali-phy {ITS1,5.8S,ITS2}.fasta -S 1,3:gtr -I 1,3:rs07 -S 2:tn93 -I 2:none --scale=1,3: --test
```

This means that the branch lengths for partitions 1 and 3 are the same, instead of being independently estimated.

Take a look at the model parameters. There is no longer a Scale[3] parameter. Instead, partitions 1 and 3 share Scale[1].

11. Dataset preparation

11.1. Splitting and merging Alignments

You might want to split concatenated gene regions:

```
% cd ~/alignment_files/examples/ITS-many/
% alignment-cat -c1-223 ITS-region.fasta > 1.fasta
% alignment-cat -c224-379 ITS-region.fasta > 2.fasta
% alignment-cat -c378-551 ITS-region.fasta > 3.fasta
```

Later you might want to put them back together again:

```
% alignment-cat 1.fasta 2.fasta 3.fasta > 123.fasta
```

11.2. Shrinking the data set

You might want to reduce the number of taxa while attempting to preserve phylogenetic diversity:

```
% alignment-thin --cutoff=5 ITS-region.fasta -v --preserve=Csaxicola420 > ITS-region-thinned.fasta
```

This removes sequences with 5 or fewer differences to the closest other sequence.

- The `-v` option shows which sequences are removed and the distance to their closest neighbor.
- The `--preserve` option keeps specific sequences in the data set.

You can also specify that data sets should be shrunk to a specific number of sequences:

```
% alignment-thin --down-to=30 ITS-region.fasta -v > ITS-region-thinned.fasta
```

To see all options, run:

```
% alignment-thin --help
```

11.3. Cleaning the data set

Keep only sequences that are not too short:

```
% alignment-thin --longer-than=250 ITS-region.fasta > ITS-region-long.fasta
```

Remove 10 sequences with the smallest number of conserved residues:

```
% alignment-thin --remove-crazy=10 ITS-region.fasta > ITS-region-sane.fasta
```

Keep only columns with a minimum number of residues:

```
% alignment-thin --min-letters=5 ITS-region.fasta > ITS-region-censored.fasta
```