Exercises

November 30, 2007

1 Introductory remarks

General comments We have attempted to compile a set of exercises and questions that represent a selection of typical steps you will perform, and problems you might encounter, when doing phylogeny reconstruction analyses. The data you will work with is, whenever possible, 'real biological data' rather than simulated sequences, and thus as close to phylogeny reconstruction reality as possible. However, in a few cases the data does not have all the features we would like them to have. Rather than switching to a different data set then, we will ask you to modify original sequences or alignments in a way that the problem we would like to point out becomes obvious. *Please make sure that whenever you modify a file you save it under a different file name, and that you keep track of what version you are using.*

Web pages and data bases In the course of the exercise, you will need to go to several web-sites and databases to collect data for your analyses. Unfortunately we lack the time to mention every single of these web sites in detail. If you are not already familiar with these pages, please take some time and have a look what kind of information they provide. However, before you get completely lost, please ask!

Functions and programs Similar to the web pages and databases, a number of unix-functions, such as *less, chmod, sed, grep, head, tail, tr* might be handy to use for data manipulation and quick data analysis. Again, we will not be able to give a thorough introduction into every single one of these functions. Rather, we mention them in appropriate places and leave it to you to read the man-pages and find out how to use them in case you are not already familiar

with them. Please keep in mind that sometimes it's worthwhile to read and think for an hour finding out how a program can do something for you in less than a second, even though you would only need five minutes to do it manually. Again, we can only suggest how to accomplish certain tasks. If you find other ways to be more efficient, go for it.

Documentation and solutions One part of the exercise is the documentation of the individual steps you have done during your analyses. Please enter the answers to the individual questions you'll find below at the appropriate place in your documentation. Furthermore, please add a remark to the individual questions, whether you find them

- trivial
- appropriate
- complex
- too difficult

We will collect your documentation at the end of the course and will use it for our own evaluation of the lecture. So please make sure that you either do have it in electronic format or keep a hard copy for your own record.

2 Sequence retrieval and sequence alignment

The first set of exercises and questions is concerned with putting together an initial data set for phylogeny reconstruction

2.1 Collecting a dataset

1. ! Download the sequences gorgo-genomic.fa and ponpy-genomic.fa from the following location:

   http://www.cibiv.at/~ingo/Brno/exercises

   Note, the sequences are in *fasta-format*. In this format, each sequence has to be preceded by a sequence header starting with a >. Any information can be stored in this sequence header. The actual sequence starts in the next line following the header. Both, protein and DNA sequences can be

stored in fasta format. Individual sequences, multiple unaligned sequences and alignments can be stored in fasta format.

(a) What type of biological sequence is it?

(b) How long are the sequences?

(c) How many undetermined positions (N) do they each contain?

2. ! Try to change the format of gorgo-genomic.fa into:

   (a) Phylip

   (b) Nexus

   Please use **clustalw** for this purpose. Try *clustalw -h* for online help. Focus on the options for **DATA, VERBS, PARAMETERS**. Describe the result of your attempt and why you get this result.

3. Perform a BLAST search with gorgo-genomic.fa at

   `http://www.ncbi.nlm.nih.gov/blast/Blast.cgi`

   Run two data base searches, one BlastN search and one BlastX search to get more information about your sequence. Choose in both cases 'nr' as the search database.

   (a) What is the difference between BlastN and BlastX searches?

   (b) Why is the maximal hit length in BlastX substantially shorter than in the BlastN search?

   (c) What can you say about the information content of gorgo-genomic.fa

   (d) What would be your first guess about the species from which gorgo-genomic.fa was determined from?

4. ! Determine the corresponding regions to gorgo-genomic.fa from the following species:

   - Homo sapiens (homsa)
   - Pan troglodytes (pantr)
   - Macaca mulatta (macmu)

To do so, please go the Human Genome Browser Website at:

http://www.genome.ucsc.edu

and choose **Blat** from the menu list. Perform the Blat-search on the three genomes and:

(a) download the corresponding dna-sequences. Name them homsa-genomic.fa, pantr-genomic.fa and macmu-genomic.fa, respectively.

(b) Change the sequence header to contain only the short name of the species.

(c) Put the sequences of all 5 taxa into a single multi-fasta file. Name this file dataset1-genomic.fa

5. ! Align the sequences in dataset1-genomic.fa with clustalw.

   (a) Make yourself familiar with the alignment you have just generated.

   (b) Create a copy of the file dataset1-genomic.fa and name it dataset1-genomic-trunc.fa.

   (c) Modify the un-aligned file dataset1-genomic-trunc.fa in the following way: in the human sequence (homsa) delete the subsequence starting with *AGGAGAA* and ending with *ATCAGCTT*. This shortens the sequence by 300 bp from 837 bp to 537 bp.

   (d) Align the sequences in dataset1-genomic-trunc.fa with clustalw. Compare the alignment result with the alignment of the unmodified sequences. Does the alignment reflect your expectation? What effect to you observe?

   (e) Store the alignments for dataset1-genomic.fa and the clustal-alignment for dataset1-genomic-trunc.fa for later analyses in all three file formats: *fasta, nexus, phylip*.

6. ! Gather a second data set that covers now a broader phylogenetic range. Start with the human gene identified by **ENSG00000142687**.

   (a) What does the gene code for?

(b) Get the corresponding proteins and transcrips (*orthologs*) for the following species:

- *Homo sapiens*, Human, HOMSA
- *Pan troglodytes*, Chimp, PANTR
- *Macaca mulatta*, Rhesus, MACMU
- *Mus musculus*, Mouse, MUSMU
- *Rattus norvegicus*, Rat, RATNO
- *Canis familiaris*, Dog, CANFA
- *Takifugu rubripes*, Pufferfish, FUGRU
- *Ciona intestinalis*, Sea squirt, CIOIN

(c) Put the protein sequences into a multi-fasta file called dataset1-protein.fa, and the transcript sequences into a multi-fasta file called dataset1-transcript.fa.

(d) Align both files with clustalw and create nexus and phylip versions of the alignments.

(e) Why do we need to use orthologous sequences for species tree reconstruction?

To complete this exercise, please go to the the following web site to obtain the sequences:

http://inparanoid.sbc.su.se

Use **Search by Sequence IDs** and the Gene id provided above. Note, in some format, such as nexus or phylip, the length of the sequence name is limited to 10 characters and will be truncated if they are longer. It is, thus, helpful to restrict sequence names to 10 characters right from the beginning.

3 Phylogeny reconstruction using Maximum Parsimony (MP) and Sequence distance

By now, you have gathered three sets of orthologous sequences, that can be used for tree reconstruction. Furthermore, you should have one modified alignment

that will serve as an example of how artefacts introduced by errors in the analysis can influence your conclusions. If you haven't had sufficient time to complete all of the previous exercises, you can download the missing datasets from

http://www.cibiv.at/~ingo/Brno/exercises/datasets/

However, please make sure to complete the exercises whenever you have spare time. In the following, you will use various simple methods to reconstruct the phylogeny of the taxa under study.

3.1 The criteria of Maximum Parsimony

Analysis by eye

1. ! Open the alignment file dataset1-genomic.aln in text editor. Identify and count the number of parsimony informative sites in the alignment.
2. ! What tree does this alignment support. Write the tree
   (a) in Newick-format
   (b) as a graph
3. Check the results of the previous exercise by trying to open the tree file in newick-format with a tree viewer software.
4. How many unrooted tree topologies exist for 5 taxa?
5. ! Assign the parsimony informative sites to the individual splits in the tree. Do all parsimony informative sites argue for the same tree? If not:
   (a) Which site disagrees?
   (b) What split does it support?
   (c) What could be reasons for the observation of such incompatible splits?
6. Given the tree, what species would you assign the two unknown sequences to? Compare your assignment to the conclusion you have drawn from the Blast-search.

Analysis with software tools: The program *paup* Since *paup* is a commercial software, we need to run the analyses on the embnet-cluster in Vienna. Please log in via ssh into the emb-net server. To do so, type

ssh <your-account>@emb1.bcc.univie.ac.at

Enter your password. Next, transfer the four alignments to your home directory using *scp*. *paup* requires the alignment files to be in nexus format! To do so, issue the following command in a second shell:

scp <yourfilename> <your-account>@emb1.bcc.univie.ac.at:

Please check next, if the file transfer was successful. For obtaining a quick-start introduction of *paup* please visit the following link:

`http://www.cibiv.at/~ingo/Brno/paup_quick-start.pdf`

1. ! Open the file dataset1-genomic.nxs in *paup*. To do so, simply type *paup dataset1-genomic.nxs*

2. ! Type help to get an overview of the options *paup* gives you.

3. ! What information does the command *cstatus* give you? Compare this information to the results you have obtained in your analysis of the alignment by eye.

4. ! Set the information criterion to parsimony. Compute the maximum parsimony tree using the command *alltrees*. What kind of search do you perform? How many trees are evaluated?

5. ! View the result of your tree search by issuing the command *showtrees*. Compare the result to your manually reconstructed tree. Do they agree?

6. ! Compute the maximum parsimony tree using the command *hsearch*. Make sure to understand the difference to the command *searchall*?

7. Set **homsa** (*Homo sapiens*) as the outgroup, root the tree and display it. Next, unroot the tree, change the outgroup to **macmu** (*Macaca mulatta*) and re-root the tree. Hint: use the commands *outgroup, roottrees, deroottrees*.

8. Save the tree for later inspection.

3.2 The criterion of distance

1. ! Load the alignment dataset1-genomic-trunc.nxs into *paup*. Use the command *execute*. Remember, that these are the same sequences as in the alignment dataset1-genomic.nxs, with the exception that you have removed 300 bp from whithin the human sequence. Compute the maximum parsimony tree and compare it to the tree from the previous exercise. What are the differences.

2. ! Compute the neighbor joining tree for the alignment. To do so, use the command *nj*

3. ! Compute the optimal tree according to the distance criterion. Set the criterion to distance, and perform both exhaustive and heuristic search. Compare the distance tree to the MP-tree. Explain your observation.

4. Close *paup*, activate the t-coffee environment (simply type *activate T-COFFEE* and perform a t_coffee alignment of dataset1-genomic-trunc.fa. To do so, simply type *t_coffee* <*filename*>. Reformat the output to nexus and repeat the tree reconstruction using MP and Distance methods. Compare the results to the ones obtained with the clustalw-alignment. Interpret your results.

3.3 Robustness of the tree reconstruction inferred from resampling methods

1. ! Load the dataset dataset1-transcript.nxs into *paup*. Set the criterion to parsimony. Perform bootstrap and jackknife analysis with default settings on the tree (make sure that you understand what these two methods do). What can you say about the support of the individual splits in the tree?

2. Switch to distance analysis and repeat the analysis. Compare the results to the MP-analysis. Why do these two criteria differ in the robustness of the reconstructed tree?

3. Repeat the bootstrap and jackknife analysis with the alignment dataset1-protein.nxs. Why does MP perform better than with the DNA sequence data?

4. ! Download the sequences drome-protein.fa, aedae.fa and anoga.fa from

   http://www.cibiv.at/~ingo/Brno/exercises/dataset1/protein/

   Add these sequences to the sequences in dataset1-protein.fa, perform a clustalw-alignment, convert to nexus format and perform Distance-tree reconstruction.

5. ! Do the three sequences form a monophyletic clade?

6. All sequences that you have analyzed so far, with the exception of *Ciona intestinalis*, are vertebrates. By looking at the tree, can you decide whether the three new sequences are from vertebrate species or not? Furthermore, by looking at the tree, can you decide whether the three sequences come from different individuals of a single species, or from three different species? Explain.

7. Visualize the tree using the command *describetrees*. Note, you need to specify what tree should be described. Try the command *describetrees ?*.

8. What is the branch length of the *Takifugu rubripes* branch?

4 Maximum Likelihood with Phylip

All of the following exercises will be using the package phylip. Phylip is not a single program, but rather a group of programs that all solve smaller problems. For example the program `dnaml` finds the ML tree from DNA sequences while `proml` uses protein sequences. You can find a rather large introduction at:
http://www.cibiv.at/~greg/Phylip/phylip2.pdf.
The Web page is:
http://evolution.genetics.washington.edu/phylip.html
Phylip, like many other programs, uses slightly different file formats. You can use clustalw to convert alignments from one format to another. We will need to convert clustalw alignment files to phylip files. This can be done as follows:

```
clustalw -convert -output=phylip -infile=in.aln -outfile=out.phy
```

Phylip also tends to be very sensitive with file formats. The real problem is that when a file format is wrong it will not tell you, but the program will simply crash. So if something is not working it is very likely the file format.

Phylip writes its output to the same file names, regardless of what the program does. So read the warnings and remember to move/copy output files before running other phylip programs.

4.1 Short Phylip command List

Command	description
dnaml	Finds a ML tree from DNA sequences
dnamlk	same as dnaml only with a molecular clock
proml	Finds a ML tree from Amino Acid sequences
seqboot	generates bootstrap replicates
treedist	Compares trees
retree	allows tree manipulation i.e rerooting
consense	combines a set of trees into a consense tree

4.2 Small trees

1. Using the alignment file dataset1-genomic.aln from the previous exercise. Convert the file to phylip format, then use `dnaml` to find a ML tree. Do this with global rearrangements on and off as well as with both the rough method on and off.

2. There is a option to Randomise input order of sequences. Why do you think this is provided. Does it change anything? Don't forget that dnaml will overwrite old output files. Also in order to compare trees you can use the `treedist` program. However these are small enough to just look.

3. Compare the resulting tree with the Maximum Parsimony tree and comment on differences. Would you expect there to be large differences?

4. Find the ML tree with the truncated sequence data (dataset1-genomic-trunc.aln) and compare to original tree. Do the options change any details?

4.3 Larger Trees, and Bootstraps

When using rate variation models, use a Coefficient of variation of substitution rate among sites of 1. When you are asked about HMM categories, use 4. This is just the number of categories in other programs.

1. Using the dataset1-transcript.aln dataset, find the ML tree with dna sequences.

2. Again use global rearrangements and randomise the input sequence order. Does this change the tree? If so, why?

3. Use the gamma model for rate variation among sites. What is different about the tree? Consider branch lengths when comparing.

4. Does the result change much when the transition transversion ratio is changed? Explain.

5. Use the program `seqboot` to generate 100 re-sampled dataset replicates.

6. Find the ML trees for all 100 datasets by using the "Analyze multiple data sets?" option in dnaml. Do not use rate variation across sights.

7. Find the consense tree of the 100 bootstrap samples with `consense`. Use both the MRe and the strict option if appropriate.

8. Repeat the bootstrap analysis while including rate variation. Which gives better bootstrap support? Why do you think this is the case?

9. Carry out a jackknife again by using seqboot to generate 100 replicates. Compare the support with a normal bootstrap.

10. Now use the Amino Acid sequence data from dataset1-protein-ext.aln. Using global and non-rough options find the ML tree `proml`. Compare to the DNA based tree.

11. Estimate the tree with Rate variation. Compare to previous results.

12. Carry out a normal bootstrap and a jackknife on the protein sequences in the same way as before with the protein sequences. How does the support compare to the DNA bootstrap and jackknife results.

4.4 Extra Exercises

1. Estimate ML trees from both dataset1-transcript.aln and dataset1-protein-ext.aln using a different ML program. IPQNNI and phyml and Tree Puzzle are all examples. Phyml has a web site http://atgc.lirmm.fr/phyml/

and Tree Puzzle is installed locally. Compare to previous results and explain why there are difference if any.

2. Carry out bootstraps with the program that you chose in the previous question. How do the results compare to the previous results and also how do the DNA to protein tree compare?