Workshop practical on internode certainty and related measures

Jacob L. Steenwyk & Antonis Rokas

Programs that you will use: RAxML, IQ-Tree, Python, R

Internode certainty and related measures (Salichos and Rokas 2013; Kobert et al. 2016) have proven to be powerful methods to examine bipartition support. This has proven to be especially true when bootstrap values become unreliable (e.g., when an input alignment is very long).

In this practical, we will learn how to calculate internode certainty and related measures to examine bipartition support and support for alternative topologies. We will be examining the relationships between filamentous fungi known to be medically and technologically significant. More specifically, we will be examining the relationships among *Aspergillus*, *Penicillium*, *Monascus, Xeromyces, Penicilliopsis*, and outgroup taxa (*Uncinocarpus reesii* and *Coccidioides posadasii*) using phylogenies from Steenwyk et al. 2018.

Note, steps within objectives that have fill-in-the-blank prompts are indicated as such using blue color font. Please fill in these prompts.
Additionally, if the software you are trying to use isn't in your path, it is likely in *~/software*

Protocol

1) Download and examine the dataset

Objectives:

i) Download the directory and examine the contents.

2) Prepare the necessary input files and calculate internode certainty and related measures

- The necessary input files to calculate internode certainty using *RAxML* include the putative species tree and evaluation trees.
- The species tree has been provided and is named *Asp_Pen_phylo.subset.new*. It contains a subset group of eleven taxa from the original 82 taxa dataset (Steenwyk et al. 2018).
- Evaluation trees can be from bootstrap replicates or single gene phylogenies; the latter of which is more common. In either case, the file containing evaluation trees will have one phylogeny in newick format for each line of the evaluation tree file. The evaluation trees are a subset from the original 1,668 data matrix.

Objectives:

i) How many single gene trees are in *FILES_AA_trees_subset?*

ii) Create a file of evaluation trees with the following command:

```
cat FILES_AA_trees_subset/* > 1287.trees
```

iii) Calculate values of internode certainty and related measures using the following command:

```
raxmlHPC -f i -t Asp_Pen_phylo.subset.new -z 1287.trees -m GTRCAT -n Asp_Pen_phylo.ic -C
```

where *-f i* specifies that calculations of internode certainty, internode certainty all, tree certainty, and tree certainty all will be displayed on a specified input tree. The input tree is specified with the *-t* parameter. *-z* species the evaluation trees, *-m* specifies an arbitrary substitution model (i.e., it has no influence on the results but is a required argument for *RAxML* to work), *-n* specifies a string given to output, and *-C* will provide a verbose output.

The verbose output provides the user with two additional file types: (1) files with the reference and alternative topologies for each bipartition, which is specified in the following syntax of *RAxML_verboseIC.Asp_Pen_phylo.ic.0 … RAxML_verboseIC.Asp_Pen_phylo.ic.N-1* where *N* is the number of bipartitions in a given tree and (2) a summary file, *RAxML_verboseSplits.Asp_Pen_phylo.ic*, of various topologies, gene support frequencies, and internode certainty values for each bipartition. Other output files include an information file, *RAxML_info.Asp_Pen_phylo.ic,* which contains details about the *RAxML* run, and *RAxML_IC_Score_BranchLabels.Asp_Pen_phylo.ic,* which contains the input tree with internode certainty and internode certainty all values on the tree represented as branch labels.

iv) Examine the contents of the information file, *RAxML_info.Asp_Pen_phylo.ic*. Determine the tree certainty and relative tree certainty value using the following commands:

```
grep "Tree certainty\|Relative tree" RAxML_info.Asp_Pen_phylo.ic
```

where *grep* searches for the string *'Tree certainty'* and *'Relative tree'* in *RAxML_info.Asp_Pen_phylo.ic*. In *'Tree certainty\|Relative tree'* of the *grep* command, the *'\|'* specifies to search for the string *'Tree certainty'* and *'Relative tree'* thereby allowing you to search for multiple strings in one command.

Fill in the following values:

Tree certainty value:

Relative tree certainty:

Tree certainty all including all conflicting bipartitions:

Relative tree certainty all including all conflicting bipartitions:

3) Identify bipartitions with low internode certainty values

• To examine internode certainty values among bipartitions in our putative species tree, we will first execute a script that replaces bootstrap values with internode support values and forgo examination of internode certainty all values. We will then view the tree in *FigTree* or similar software.

Objectives:

i) To report internode certainty values as support values, we will use the file *FILES_scripts_and_matrices/report_ic_as_internode_support.py*. To understand how to use the script, execute the following command:

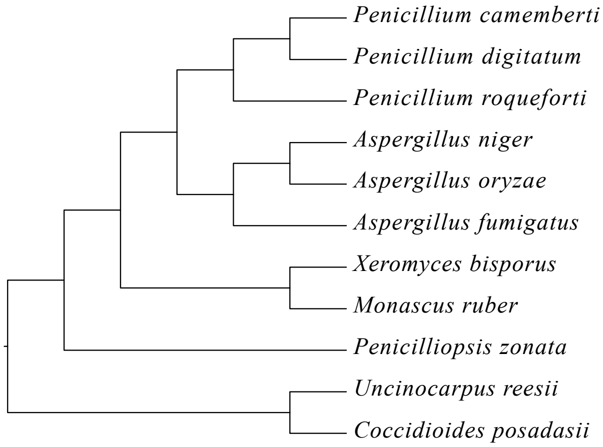
```
python FILES_scripts_and_matrices/report_ic_as_internode_support.py -h
```

ii) Based on the help message printed from the previous command, execute the script on the following tree *RAxML_IC_Score_BranchLabels.Asp_Pen_phylo.ic* to report internode certainty values as support values.

iii) Open up the resulting file, *RAxML_IC_Score_BranchLabels.Asp_Pen_phylo.ic.ic.tree*, using *FigTree*.

iv) In Steenwyk et al. 2018, we further scrutinized bipartitions with internode certainty values <0.1. Our reasoning is that internodes with this value or lower have ~5 evaluation trees that support the reference topology and ~2 or more phylogenies that support an alternative topology - we consider this substantial incongruence.

v) Identify the internode with an internode certainty below 0.1 and specify the value and which internode that is in the phylogeny below (see next page).



4) Determine gene support frequency and concordance factors for the reference and alternative topology at the contentious internode

- To determine gene support frequency, we will use *RAxML*.
- To determine concordance factors, we will use *IQ-Tree*.

Objectives:

i) Examine the verboseSplits output from *RAxML*, which is file *RAxML_verboseSplits.Asp_Pen_phylo.ic*

ii) The bipartition for the contentious internode of question is second to last instance of the string *'partition:'*, which appears as the following:

```
partition:
---** ***** *      xx/yy/zz
--*-- ***** *      xx/yy/zz
--*** ----- -      xx/yy/zz
```

where *xx/yy/zz* represents the following:

xx: the number of genes that support a given bipartition,

yy: the gene support frequency of the given bipartition, and

zz: the internode certainty value for the given internode.

We will focus on the first two topologies which consider whether genera *Penicilliopsis* is more closely related to outgroup taxa or if *Xeromyces* and *Monascus* is. In other words, which genus/genera *Penicilliopsis* or *Xeromyces* and *Monascus* is sister to all other members of this fungal family?

What is the number of genes and gene support frequency that support the two topologies?

First topology:

Second topology:

iii) **Note**, there is a bug in *IQ-Tree* beta version 6, which is installed on the instances. Therefore, you will have to install beta version 7 where the bug has been fixed and you will therefore be able to calculate concordance factors. To do so, execute the following series of commands:

```
wget https://github.com/Cibiv/IQ-TREE/releases/download/v1.7-beta7/iqtree-1.7-beta7-Linux.tar.gz
```

```
tar -zxvf iqtree-1.7-beta7-Linux.tar.gz
```

Now, calculate gene and site concordance factors in the following single command.

```
iqtree-1.7-beta7-Linux/bin/iqtree -t Asp_Pen_phylo.subset.new --gcf 1287.trees --scf 100 -p FILES_AA_fastas_subset/ --prefix gcf1297_scf100_ref
```

where *-t* specifies the putative species tree, *--gcf* represents the *1287.trees* observed single gene trees, *--scf 100* represents that 100 quartets will be randomly for each internal branch for computing site concordance factors, *-p* specifies the directory that has the single gene fasta files, and *--prefix* specifies the prefix of the output files.

There are four resulting files: (1) a *log* file, (2) a *cf.tree* file with gene concordance factors, (3) a *cf.branch* file with internal branch identifiers, and (4) a *cf.stat* file with tab-separated values of gene and site concordance as well as gene and site discordance factors for each internode. For now, we will focus on gene and site concordance factors.

iv) Reexamine the contentious bipartition in the *gcf1297_scf100_ref.cf.tree* file, which can be viewed in *FigTree* or similar software.
What is the gene concordance and site concordance factor for the contentious bipartition?

v) Conduct the same analysis using the alternative topology as an input by executing the following command:

```
iqtree-1.7-beta7-Linux/bin/iqtree -t Asp_Pen_phylo.subset.alt.new --gcf 1287.trees --scf 100 -p FILES_AA_fastas_subset/ --prefix gcf1297_scf100_alt
```

vi) Examine the contentious bipartition where *Penicilliopsis zonata* splits from *Aspergillus* and *Penicillium* in file *gcf1297_scf100_alt.cf.tree*. What is the gene concordance factor and site concordance factor for this bipartition?

5) Calculate gene-wise and site-wise phylogenetic signal

- To determine gene-wise and site-wise phylogenetic signal, we will follow protocol first described in Shen et al. 2017 and calculate gene-wise and site-wise log-likelihood scores. We will use *IQ-Tree* to facilitate the analysis.

Objectives:

i) Calculating gene-wise and site-wise log-likelihood scores takes too much time – **so do not execute the following commands**. The following commands are provided because I wanted you to have access and an understanding of how the results files you will be using were created.

Command 1:

```
iqtree -s FILES_scripts_and_matrices/concat.fa -seed 78913467814 -st AA -pre
likelihood_ref -spp FILES_scripts_and_matrices/partition.file -te
Asp_Pen_phylo.subset.new -wpl -wsl
```

Command 2:

```
iqtree -s concatenation.fasta -seed 13469781343 -st AA -pre likelihood_alt -spp
partition.file -te FILES_GLS_files/concatenation.alt.topology -wpl -wsl
```

ii) In the previous two commands, what do the following specify? Hint, it will be useful to use their documentation – see http://www.iqtree.org/doc/Command-Reference.

-spp:

-te:

-wpl:

-wsl:

iii) Create summary files with gene-wise and site-wise log likelihood scores using a custom script provided for you. To do so, first change directories into *FILES_gene_and_site_lh_values* with the following command:

```
cd FILES_gene_and_site_lh_values
```

Next examine how to use the script that will create summary files for you using the following command:

```
bash ../FILES_scripts_and_matrices/create_GLS_summary.sh -h
```

Based on the help message, create gene-wise and site-wise log likelihood score summary files using the following command:

```
bash ../FILES_scripts_and_matrices/create_GLS_summary.sh
../FILES_scripts_and_matrices/partition.file likelihood_ref.partlh
likelihood_alt.partlh likelihood_ref.sitelh likelihood_alt.sitelh
```

iv) We will now examine the contents of *gene-wise_logli_scores.txt* and *site-wise_logli_scores.txt.* To do so, we will plot the results from each file in *R* using *ggplot2*. First, start a session in *R* by typing *R* in the terminal and load *ggplot2* using the following command:

```
library(ggplot2)
```

where *library()* is used to load the *R* package *ggplot2*. If you are not in the working directory with your results files, use *setwd('path')* to change to the appropriate directory. Make sure you are in the appropriate directory with the *getwd()* command.

v) Next, read in the data tables using the following command:

```
gene<-read.table("gene-wise_logli_scores.txt", sep = "\t")
```

where *read.table()* is an *R* function to read in a table and *sep = "\t"* specifies that the table is tab (or *"\t"*) delimited.

vi) Make a plot of the gene-wise log likelihood scores using the following command:

```
ggplot(gene, aes(V1, V4, color=V5)) + geom_bar(stat="identity") + ggtitle("GLS
(gene-wise)") + xlab("Genes") + ylab("deltaGLS") + theme_classic() +
theme(axis.text.x=element_blank(), axis.ticks.x=element_blank()) +
scale_color_manual(values=c("#56B4E9", "#999999", "#E69F00"))
```

vii) There is one gene with overwhelming phylogenetic signal. To determine what gene this is, sort the data stored as *gene* according to the column of ΔGLS values using the following command and examine the fourth column, which is ΔGLS:

```
head(gene[order(-gene$V4),])
```

viii) What gene has a high ΔGLS in favor of the reference topology?

ix) To determine if removing this gene will switch the topology at the given internode, calculate the sum of log likelihood values after removing the gene-wise log likelihood score for this gene. To do so, execute the following command:

```
sum(gene[(gene$V4!= 113.180),]$V4)
```

where *sum()* will calculate the sum of a set of numbers and *gene[(gene$V4!=113.180),]$V4* will print the set of numbers excluding the highly opinionated gene.

x) Does the topology change if you remove this gene?

xi) Repeat v and vi for *site-wise_logli_scores.txt* by executing the following commands:

```
# read in site-wise log likelihood scores file
site<-read.table("site-wise_logli_scores.txt", sep = "\t")
```

```
# create plot
ggplot(site, aes(V1, V4, color=V5)) + geom_bar(stat="identity") + ggtitle("SLS (site-wise)") + xlab("Sites") + ylab("deltaSLS") + theme_classic() +
theme(axis.text.x=element_blank(), axis.ticks.x=element_blank()) +
scale_color_manual(values=c("#56B4E9", "#999999", "#E69F00"))
```

xii) Is there one site with overwhelming signal?