Welcome page

What is a Wiki Site?
How to edit pages?

How to join this site?
Site members

Recent changes
List all pages
Page Tags

Site Manager

Page tags

start

Add a new page

new page

edit this panel

Advice On Statistical Model Comparison In Biogeobears

Statistical estimation of models via maximum likelihood (ML), statistical model choice using the likelihood ratio test, AIC, AICc, etc., have become ubiquitous in phylogenetics and many other fields, but they are still somewhat new in historical biogeography. Here I summarize of the basic principles and address a few common misunderstandings.

I do not introduce any equations here, rather, my goal is to briefly explain the terminology and concepts as they are used in discussions of BioGeoBEARS.

NOTE: I have also added an Excel spreadsheet, with the basic AIC, AICc, and LRT calculations done by hand, to compare to an example BioGeoBEARS output. See "Files" at the bottom of the page, or this link: http://phylo.wikidot.com/local--files/advice-on-statistical-model-comparison-in-biogeobears/LRT_AIC_tables_v2.xlsx

Terminology

(link to this section)

Here are the basic terms used in discussions of likelihood-based estimation and model-testing.

- **Likelihood** = P(data|model) = probability of the data, given a model. Note that this is a technical definition and not identical with colloquial understandings of the word "likelihood".
- **LnL** = log of the likelihood. Note this is always the NATURAL log.
- **Maximum Likelihood (ML)** = a statistical technique for estimating model parameters by finding those parameters that maximize the probability of the data under the model (the likelihood). For simple problems, the ML solution can be found analytically by taking the equation that gives the likelihood as a function of a parameter, taking the derivative of the equation, and solving for 0 (which represents the point where the slope is flat, presumably the maximum of the curve). For more complex problems, an iterative "hill-climbing" routine is used to find the likelihood peak. There are a wide variety of such algorithms, and some will work better on some problems than others.
- **Parameter** = A number in the likelihood equation which may vary (if it's a free parameter) or not (if it's a fixed parameter).
- **Data** = Observations, which do not vary once they are collected.
- **AIC** = Akaike Information Criterion, a likelihood-based measure of model fit that penalizes more complex models (more complex = more free parameters).
- **AICc** = Akaike Information Criterion, with correction for sample size (corrected AIC, or second-order Akaike Information Criterion)

Higher LnL corresponds to higher data probability, so **higher** (less negative) LnLs are better, representing better model fit to the data. LnL = -2 is better than LnL = -5.

To convert LnL to plain likelihood, take e^{LnL}. In R, this is the `exp()` function; see script below.

For calculation of AIC etc., see Brian O'Meara's webpage and references therein: http://www.brianomeara.info/tutorials/aic

Note that with AIC, AICc, etc., **lower** is better. Lower AIC indicates better model fit. So, AIC = 10 is better than AIC = 100.

Data have likelihood, not models

It is fairly common to hear researchers talking about the "likelihood of models", or "model A has higher likelihood than model B". While this is not completely horrible, it is not technically correct, and can cause far-reaching confusions in studenthood and beyond. Technically, **data have likelihood, and models confer likelihood on data.** That is, given a particular probabilistic model and model parameter values, there is a certain probability of producing the data you have observed. **The likelihood is the probability of the data given the model, NOT the probability of the model, given the data.** To get the latter, you need to use Bayes' Theorem and a Bayesian analysis.

AIC or AICc can be used to calculate model weights and relative likelihoods, which do give a sense of which models are better supported by the data. However, here you are effectively assuming equal prior probability of all of the considered models, and a zero prior probability on any models not considered.

Likelihood and log-likelihood (R script)

Short R script to show how to convert between LnL and likelihood

```
# Convert a log-likelihood to a plain likelihood/probability:
LnL = -2
likelihood = exp(LnL)
likelihood

# These are the same, showing that the default of log() is base e
log(likelihood)
log(likelihood, base=exp(1))

# exp(1) equals e:
exp(1)

# Compare two likelihoods
LnL1 = -2
likelihood1 = exp(LnL1)

LnL2 = -5
likelihood2 = exp(LnL2)

likelihood1
```



```
# You should get:
# > likelihood1
# [1] 0.1353353
# > likelihood2
# [1] 0.006737947

# You can see that #1 is higher than #2, whether you are looking
```

Basics

(link to this section)

Goal

(link to this section)

The goal of statistical model comparison is to compare the fit of DIFFERENT models to the SAME data. This means that likelihoods, AIC values, etc., can only be compared on the SAME data. Comparisons of these values across different data have no meaning. This is because likelihood means "probability of the data given a model".

(You can, of course, say something like "DEC+J is better than DEC on dataset 1, and DEC is better than DEC+J on dataset 2." You just can't conclude anything from "DEC+J confers LnL -10 on dataset 1, DEC+J confers LnL -100 on dataset 2." Dataset 2 could have lower likelihood just because it is a bigger dataset, for instance. A specific sequence of 100 coin flips has a lower probability than a specific sequence of 10 coin flips.)

Nesting of models

(link to this section)

Model A is nested inside Model B when fixing a parameter in Model B results in

a model identical to Model A. For example, in BioGeoBEARS, DEC has two free parameters, *d* and *e*, and the parameter *j* is fixed to 0. DEC+J has three free parameters, *d*, *e*, and *j*. When *j*=0, DEC+J reduces to DEC. So DEC is nested inside DEC+J, DIVALIKE is nested inside DIVALIKE+J, and BAYAREALIKE nests inside BAYAREALIKE+J.

Any other model comparisons are not nested — e.g., DEC is not nested inside DIVALIKE+J, even though DEC has 2 free parameters and DIVALIKE+J has 3 free parameters. The non-nesting occurs because these models have different fixed assumptions (different fixed parameter values, in BioGeoBEARS) controlling differences in e.g. vicariance, subset sympatry, etc.

Likelihood Ratio Test (LRT)

(link to this section)

When two models are nested, the Likelihood Ratio Test (LRT) can be used to test the null hypothesis that the two models confer the same likelihood on the data. This test is just a chi-squared test, with the test statistic (D) being 2*

p-value is CHISQ.DIST.RT.

The LRT can only be used in pairwise fashion, to compare two models. If you have a larger number of models, you could do a LRT on each pair of models (as long as one of these models nests within the other). Note that with this strategy you have to start worrying about multiple-testing bias.

Akaike Information Criterion

(link to this section)

AIC and AICc can be used to compare two models (or any number of models), whether they are nested or not. There is no statistical theory to identify a strict *p*-value cutoff for significance when using AIC or AICc. Instead, AIC/AICc give a measure of relative model probability. See:
http://en.wikipedia.org/wiki/Akaike_information_criterion and:
http://www.brianomeara.info/tutorials/aic

Maximum Likelihood (ML)

(link to this section)

ML optimization routines and their pitfalls

(link to this section)

The LRT, AIC, and AICc all assume that you have actually found the Maximum Likelihood (ML) solution, that is, the model parameter values that confer the maximum likelihood on the data. Functions to search the parameter space for the values that maximize the likelihood of the data are well-developed (see the optim() function in R base, and optimx R Package).

However, these optimization algorithms do not always work perfectly, particularly when the likelihood surface is very flat. This can occur when a model is a very poor fit, or when your starting values for a search are far from

the ML values, or when you have accidentally set the bounds of your parameter search such that the ML values of the parameters are outside the bounds. (In the latter case, if you are lucky, the ML search will keep hitting this limit, indicating that there might be higher likelihoods attainable above the bound.)

Another problem can occur when two parameters are non-identifiable — you will get a "likelihood ridge", and the true values of the parameters might be anywhere along this ridge.

All of these issues get worse as more and more free parameters are added. In BioGeoBEARS, so far we have just been using models with 2 and 3 parameters, and optim/optimx seem to work well in general.

The one optimization problem I have sometimes noticed (in perhaps 1% of datasets) are cases where the +J model has a lower likelihood under ML search than the corresponding 2-parameter model where j has been fixed to 0. If you see a 3-parameter model with lower likelihood than a 2-parameter that nests inside it (the nesting is crucial), you have immediate evidence for a problem in optimization, because the 3-parameter model should always be

This situation probably occurs when the ML value of j is close to 0, but the likelihood surface for j is flat and is interacting with d and e.

The fix, however, is easy: just use the ML parameter estimates from the 2-parameter model as the starting values for the ML optimization of the 3-parameter search. I have now made this the default setup in the example script.

Double-checking ML

(link to this section)

The more general solution to ML optimization problems is to repeat the search with a variety of different starting parameter values, and see if you keep getting the same ML parameter estimates.

I have not automated this procedure, as most of the time it isn't necessary, and it significantly slows the run time (10 searches will take 10 times longer than 1). BioGeoBEARS makes it easy to change the starting values of parameters, however (the "init" column in the params_table).

This may be particularly useful as double-check, for example if reviewers ask that you double-check you are getting the ML parameter estimates.

Improving the chance of finding the true ML solution

(link to this section)

In most real-life problems (where there is no analytical solution), you can never be absolutely 100% sure that your hill-climbing ML algorithm has found the parameter values that maximize the likelihood of the data. However, there are a number steps that can be taken, if you have indications that optimization is problematic, or just if you have reason to be worried (e.g., models with >3 free parameters in BioGeoBEARS) or cautious.

None of these strategies are guarantees, of course. The most important strategy is to always look at your data and your various ML results (parameter values, LnL, ancestral states, etc.) and ask yourself if they make sense.

Strategies for improving the search for the ML parameter values include:

1. Look at the parameter values and resulting LnL as they are printed to screen during the BioGeoBEARS search. If the likelihood climbs rapidly as the parameters shift, you probably have a strong likelihood gradient. If the parameters and LnL remain stuck near the starting values, your starting parameter values may have been in a flat region of the likelihood surface, causing an optimization problem.
2. Set `speedup=FALSE` in the `BioGeoBEARS_run_object`. To speed up the `optimx`/`optim` search, I modified the `optimx`/`optim` defaults to have a higher tolerance and a faster cutoff. When `speedup=TRUE`, this can cut search times by about 50%, as often, a lot of the search time is spent bouncing around in the tiny space near the peak, estimating the 3rd, 4th, and 5th significant digits of the parameters, which are not particularly useful or relevant. Setting `speedup=FALSE` will run the `optimx`/`optim` defaults, and may fix some optimization problems, but definitely not all of them.
3. Start searches for more complex models with the ML parameter values from a simpler, nested model (see above).
4. Start searches from a variety of plausible and extreme parameter

hit the same ML solution, you can be highly confident your search is finding the ML parameters.

5. Look at the saved `optim_result` in the `results_object` and, in the help for `optimx` and `optim`, consult the "Value" description (the "Value" section describes the "value" resulting from the function, i.e. the output/results. This will indicate whether or not `optimx`/`optim` algorithm detected a problem.
6. The most thorough thing you can do is a "gridded search", where you take say, 25 possible values of each free parameter, and then calculate the likelihood of the data for each combination of these values. You can then plot the likelihood surface yourself (e.g. a 3D plot or a contour plot for a 2-parameter problem, or a colored ternary diagram for a 3-parameter search) and see where the peak is, or find ridges or multiple peaks, if those exist. Adding more parameter combinations can explore other regions of parameter space, or increase the resolution in a region of interest (e.g. near the peak). This strategy will work as long as your sampled parameter values span the actual likelihood peak in a reasonable way.

Zen and the Art of Statistical Model Selection

(link to this section)

1. The standard (ubiquitous) advice in statistical model choice is "All models are wrong, but some models are useful." (George Box)

2. In other words, even your best-fitting model is probably still wrong — at best, it is a decent approximation of the true model. At worst, it is a horrible, poorly-fitting model. If you only ever use one model, you don't know when the model is fitting relatively well and when it is fitting poorly. The main point of BioGeoBEARS was to enable the creation of different models, so that these issues became testable against datasets.

3. Although you will never know for sure if you have the "true" model (except in cases where the data has been simulated by a computer program with a known model), scientists can use model comparison and model selection procedures to at least determine which models are better than others. By designing models to include or exclude processes that we think might be

important, we can let the data tell us which models and which processes seem to be well supported / good fits to the data.

4. The models I initially set up in the example script were just chosen to imitate models/programs currently in use in the literature (see Matzke 2013, Figure 1) — DEC, DIVALIKE, and BAYAREALIKE. Instead of just running different methods, observing that the results differ, and shrugging, I advocate for the position that we should use model choice procedures to choose the models that fit the best.

The "+J" version of each model just adds founder-event speciation, a process which had been ignored in some models, probably because of the residual effects of the vicariance biogeography tradition.

This created six models, and this was a big enough step that it will probably take some time for researchers to try out the models and the model selection procedures and get a sense of their utility. But I actually designed BioGeoBEARS as a supermodel, such that specifying certain parameters can create a variety of other models. Adding constraints on dispersal, distance, changing geography, etc., adds yet more models. So, there are many more

GTR+I+gamma model (and even more recent, more sophisticated models).

(That said, new models beyond the basic 6 have not been tested much if at all, so users should be aware of possible issues, and take them as experimental until they or others have done some serious study of the performance of the new model. See "issues to watch out for", below.

Issues to watch out for

(link to this section)

1. The optimx/optim optimizers seem to find the ML solution usually on models with 2 or 3 parameters, but the searches will become slower and less reliable with more parameters. One way to check ML searches is start from different starting values and see if the inference ends up at the same peak.

2. Another test is to see if the more complex models reliably equal or exceed the likelihood of simpler models nested within them — if they don't, you've got optimization problems.

3. Some combinations of free parameters might be non-identifiable.

4. Some models might be physically absurd.

5. See also BioGeoBEARS mistakes to avoid for other sorts of conceptual mistakes to avoid.