Integrative Biology 200B *University of California, Berkeley*
"Ecology and Evolution" *Spring 2011*
by Nat Hallinan, revised by Nick Matzke

Lab 11: Testing for Clade Imbalance

Today we are going to look at several different ways of testing clade imbalance. *Mesquite* has two limitations on its ability to test for this property. It can only simulate trees that have the same number of taxa as are in your character matrix and it has no statistic to compare the bottom two nodes. Therefore, we will do what we can in *Mesquite* and then we will turn to more hands-on methods of making these calculations.

Colless's Imbalance

Colless (1982) proposed a way of measuring imbalance in a tree. It does not compare two clades but instead the overall imbalance throughout the tree. It is calculated as:

$$\frac{\sum_{i=1}^{nodes} |n_{li} - n_{ri}|}{(n-1)(n-2)/2}$$

Where n_{li} and n_{ri} are the number of taxa descended from the left and right branches of node *i* respectively, and n is the total number of taxa in the tree. *(n-1)(n-2)/2* is the maximum possible value of the sum, so that this statistic runs from 0 to 1 with 0 being perfectly even and 1 completely lateralized.

Download the *Colless_example.nex* file from the website. In Mesquite, open the stored tree and calculate Colless's imbalance by hand. Move up the tree and add up the difference between the right and left clades at each node, then divide by the denominator.

Is this value statistically significant? We have to use simulations to compare it to values from a null distribution of trees to determine if it is. Select **Analysis > New Bar & Line Chart for > Trees, Simulated Trees**. You will now be offered several different simulation options. We will discuss what these options are below. Essentially each option represents a different null distribution of possible trees. Let's start with **Uniform Speciation**. 10 is good for the tree depth. In the **Value to calculate for trees** window select **Show secondary choices** and **Coless's Imbalance**. Let's do **999** simulations for a little power. That's a pretty chart, but we want a p-value. Use the **text** tab to get actual counts and use these numbers to calculate a p-value. The p-value is the number of trees with a Colless's imbalance equal to or higher (if you want to show that the tree is particularly imbalanced) than your tree. Is it significant?

Repeat this analysis, but this time use **equiprobable trees** for your null distribution. Are the p-values similar? What effect did the null distribution have on your p-values?

Clade Imbalance

For the rest of the lab we will be trying to determine weather two sister clades are of statistically different sizes. This will be a comparison for a single tree in which one clade is

larger and has n taxa and the other clade has m (<n) taxa. We will be testing weather m is significantly different from n using a number of different null distributions.

Something to keep in mind throughout this lab is whether to do a one-tailed or a two-tailed test. A one-tailed test would be appropriate if you had a hypothesis that clades with a certain character (ie: environment or morphology) should have more taxa than clades which do not have that character, and you are comparing a pair of sister clades in which one has the character and the other does not. However, in most situations you will first have identified that one clade is larger than the other, and after the fact you will hypothesize a reason why. In this case a two-tailed test is appropriate.

Random Partition Trees

Random Partition Trees are those trees found by randomly dividing the taxa into either clade 1 or clade 2, then proceeding up the tree randomly dividing the taxa in the same way until you have a fully branching tree. Therefore there is a 50% chance of each taxon ending up in each of the bottom two clades, and it is very easy to calculate the probability that you will have a difference between clades as great as you do. It should fit a binomial distribution with one exception. A binomial distribution includes the possibility that all the coin flips end up heads or tails, but in this case if all the taxa ended up in one clade, then we would not have a node. It is easy to calculate the probability of all taxa ending up in one clade, as $(0.5)^n$. We can therefore calculate an appropriate p-distribution as:

$$\frac{p(binomial,1tailed) - (0.5)^n}{1 - 2(0.5)^n}$$

We subtract the possibility of them all ending up in one clade from 1 tail and the probability of them all ending up in either clade from the total probability.

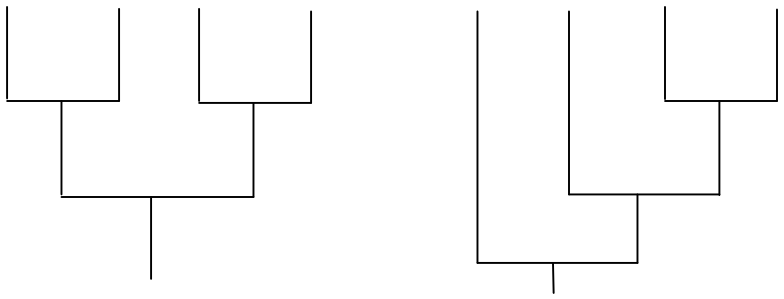
Download and open up the Excel spreadsheet labeled *Comparing_Clades.xls*. First we will calculated the one-tailed binomial distribution. In the cell labeled **binomial** type **"=BINOMDIST(B2,A2+B2,0.5,TRUE)". A2** and **B2** are the taxon totals for the big clade and small clade respectively, so that **A2+B2** is the total number of attempts. **0.5** is the probability of ending up in each clade and **TRUE** makes it calculate the cumulative binomial distribution, in other words the chance of getting a value that big or smaller.

Now calculate the p-value for our one-tailed distribution. First in the box labeled **1/2^n** type **"=POWER(0.5,A2+B2)".** In the box labeled **p-1 tail** type **"=(B6-B7)/(1-2*B7)".** Do you see how this fits the formula above? Your 2 tailed distribution is now twice that value. Mess around with the values of your big and small clades to see how they affect your p-value. What's the p-value for our tree from the previous example?

Equiprobable Trees

Another possibility is that every single labeled topology is equally likely. There is an important distinction between labeled and unlabeled topologies. Unlabeled topologies are just the trees without any taxa. While labeled topologies have the taxa assigned to the branches. Therefore, two unlabeled topologies can have different numbers of labeled topologies associated with them, and thus have different probabilities under an equiprobable trees null distribution.

For example consider the two following unlabeled topologies for four taxa. How many possible labeled topologies can you count on each one? Remember that rotating a branch does not change the topology.



One consequence of this is that imbalanced unlabeled topologies have more possible labeled topologies associated with them than balanced ones do. Thus your p-value for imbalance from a labeled distribution will be lowered relative to one that uses only unlabeled topologies.

Unfortunately I don't know how to deal with this distribution mathematically, so we're going to have to use *Mesquite* simulations to test it. In the tree window select **Taxa&Trees > Make New Trees Block From>Simulated Trees>Equiprobable**. Let's make 99. You should really make more for statistical power, but you are going to have to go over them by hand. When it offers, open the tree window.

Now open a new Excel spread sheet or just go to sheet 2 for the one that you already have opened. For each tree record in column A the number of taxa in the left clade, and record the number of taxa in the right clade in column B. If this takes forever, then screw it, but I think that it should be pretty fast.

To calculate your one-tailed p-value, in the column C subtract your value in column B from your value in column A. Now reorder all your trees according to their value in column C. Go to **Data > Sort, Expand Selection, Sort By : Column C**. You can now calculate your p-values by counting (really just use the row numbers) the number of trees with a difference greater than or equal to the tree you are comparing them to (in this case 7-2=5). Divide this number by 100 (99+1).

You can calculate your two-tailed by adding the trees counted above to those with a difference less than or equal to -5 (=2-7). Record both the one-tailed and two-tailed values. Is your two-tailed p-value twice your one-tailed? In this case any difference arises from random effects on small sample size. Therefore your 2-tailed prediction is probably better, since it is drawn from, more trees.

Shut down *Mesquite*. We're done with that.

Random Birth-Death Models

Another general model for generating a null distribution of trees is the random birth death-model. Under this model there is a constant stochastic rate of birth, usually called λ, and another constant rate of death, μ. So that, any single lineage has the same probability of speciating at any time and a different but constant chance of going extinct. These models have an advantage over the random topology models that we already discussed, because they generate branchlengths in addition to topologies.

This process is very mathematically tractable and so easy to deal with, because the rate of change for the probability that there will be n taxa in a clade at time t can be calculated as:

$$P'(n \mid t) = \lambda(n-1)P(n-1 \mid t) + \mu(n+1)P(n+1 \mid t) - (\lambda + \mu)nP(n \mid t)$$

Where *P(n|t)* is the probability of there being *n* taxa at time *t*. Although the general case is solvable we will deal with two special cases, so that the math does not get too crazy.

The first case is what happens when $\mu=0$, so that there is no chance of any taxon ever going extinct, and we expect the population to grow exponentially with time. In this case the probability of having n taxa at time t is:

$$P(n \mid t) = e^{-\lambda t}(1 - e^{-\lambda t})^{n-1}$$

Another possibility is that $\mu=\lambda$. In this case we expect the population to remain stable over time, although it's actual value may fluctuate greatly depending on the value of λ. In this situation it is very possible, even likely, for a lineage to go extinct; however, we are sampling only from lineages with extant members, so we will add a correction factor to account for the fact that we will not be observing any extinct lineages. It is important to point out that mathematically this is different from the case in which you assume that the number of lineages can not drop below 1, which is very difficult to solve for. Anyways, under this model:

$$P(n \mid t) = \frac{1}{(\lambda t + 1)}\left(\frac{\lambda t}{\lambda t + 1}\right)^{n-1}$$

It is very straight forward to generate random trees under this model. When doing so you have two choices about what to hold constant. You could hold the number of taxa at the end of the process constant, or you could hold the rate of duplication and death constant. We will now look at the effects of either assumption.

Random Birth-Death with Constant Number of Taxa

To generate random trees under an assumption of a birth and death model with a constant number of taxa, you run the process forward in time until you get a tree with n taxa. At that point you stop the process and look at the trees. This process with μ=0 is equivalent to the **Uniform Speciation (Yule)** option available in *Mesquite*. The **Uniform Speciation with Sampling** option works the same, except that it runs the process forward until there are n+x taxa and then randomly eliminates x of them.

For our purposes, one very nice thing about using a constant number of taxa is that the p-values are very easy to calculate. Under a birth-death model the chance that a clade with n taxa has r taxa on its right branch and n-r taxa on its left branch is the same no matter what the values of r, λ and μ are. Therefore, the probability of any particular break down is 1/n, and the one-tailed p-value for the imbalance is **r/n**, when r is the number of taxa in the smaller clade.

Go back to the first *Excel* spread sheet. In the cell labeled **p-1 tail** under ***Birth Death Constant Taxa*** type **"=B2/(A2+B2)".** Wow that was easy. How does that value compare to the p-value for our other models?

Random Birth-Death with Constant Rates

To generate random trees under this model you run the process forward until the total time has expired. You may end up with a different number of taxa than in your original tree. What rate should you use? An obvious choice is the maximum likelihood for your observed tree. This makes sense, because that set of values is more likely than any others to generate your observed tree.

We will assume for the purposes of this lab that the speciation at the base of your tree has already happened and we will try to maximize the likelihood for a rate that would produce a clade of size m and a clade of size n. The maximum likelihood rate depends on what value of μ you use. If $\mu=0$, then $\lambda t=\ln[(m+n)/2]$. However, if $\mu=\lambda$, then $\lambda t=(m+n)/2-1$. You would get these same results even if you considered the topologies of the two clades in question. However, if you also considered the branchlengths, it would be more difficult to calculate.

The next thing that you have to choose is what benchmark you will use. That is to say what value for the difference in size between the two clades is the basis for deciding weather the two clades in your tree are more different than the clades on the other possible trees. This was not an issue for the other methods, because all the trees had the same number of taxa, so any benchmark you chose would get the same results.

Let's start by using difference between the numbers of taxa in each clade as a benchmark. Not only is this an obvious statistic, but it is also possible to analytically solve for the p-values. As a matter of fact no matter which value of μ you choose you get the same result. The one-tailed p-value for getting a difference at least as big as n-m is:

$$\frac{m+n}{2(m+n-1)}\left(\frac{m+n-2}{m+n}\right)^{n-m}$$

So go to *Excel* and under ***Birth Death Constant Rate*** I've already typed in this formula to calculate the p-values. You just need to set the **Sum** cell to **"=A2+B2".** Do you see how all those formulas fit together to make the above formula? How do those p-values look? How do they compare to the other p-values that you've calculated? How do they change as you vary the number of taxa in clade 1 and clade2? Vary the size of the big clade a lot and look at the effects.

There is one other benchmark that can be derived and compared to a null distribution in *Excel*: the good old Likelihood Ratio Test. Making the same assumptions that we have made above, the maximum likelihood for each branch considered separately is:

$$\frac{1}{N}\left(1-\frac{1}{N}\right)^{N-1}$$

Where N is n or m depending on which clade you are considering. Type this formula in for the big clade under ***Likelihood Ratio Tests*** in the cells labeled **Big Like**. I've already typed the appropriate formula into **Small Like**. The Maximum Likelihood if you force both clades to have the same rate is:

$$\frac{4}{(m+n)^2}\left(1-\frac{2}{m+n}\right)^{m+n-2}$$

Type this formula into the cell labeled **combined**. You can now calculate the LRT as ln(big like * small like/ combined like)/2. Compare this to a chi-squared distribution with one degree of freedom to get the 1 tailed p-values **("=chidist(like ratio,1").** How do those p-values look?

Random Birth-Death Using other Benchmarks

Nat had a Windows exe program to perform the below; you can try it if you have Windows. More importantly, in the next section I have put in some R code that will generate random trees, and also will calculate some basic tree balance statistics. -- Nick

So at this point I got obsessed with what other benchmarks you could use for the random birth-death process. I came up with several ideas and wrote a program to test them. This program works by calculating a maximum likelihood λ from your data and simulating a bunch of trees, assuming $\mu=0$. For each tree it then asks if the benchmark calculated for that tree is above or below the bench mark calculated for your tree. It reports the fraction of simulated trees with a bench mark below your value. The good thing about this is that you can calculate p-values for formulas that you can't solve arithmetically.

Open the program called *Compare_Clades.exe*. Enter values for the big and small clades. As you can see, there are several different benchmarks to choose from. Let's start with the **difference**, which we already explored. Choose **type 1** error, and how ever many tails you want. You should do at least **10,000 reps**. The more reps you do, the better your estimate of p will be. Hit enter a couple of times. The program will run through the simulations and calculate a p-value. How does this p-value compare to the one calculated in our spread sheet? Repeat that analysis using the Likelihood Ratio Test. How does this p-value compare to the spread sheet p-value for the LRT? Why the difference?

Use this program and the spread sheet to calculate p-values for the random partition, the birth death with constant taxa, and the birth death with constant rate using the difference, the difference/total, the ratio and the likelihood ratio test. You also have your one calculation using equiprobable trees for 7 taxa in the big tree and 2 in the small. Fill out the p-values in the spread sheet at the bottom of the page using a wide range of values for your small clade and big clade. How do the p-values compare between these different methods? Why does the Random Partition model have such low p-values (or maybe it's better to ask why do the other methods produce higher p-vales)? Is that a legitimate measure? Does clade size have the same effect on all of them?

You can also use this program to calculate power (that is 1-type 2 error). Calculate the power of these different models (or at least the ones that he program can look at). Use these to fill in the rest of your spread sheet. How does power vary between the different methods? What about different clade sizes?

SymmeTREE

I should at least mention the program *SymmeTREE*. It claims to be able to look at an entire tree and identify if there is imbalance. It can then identify on what branches that imbalance arises. I did not have time to investigate the logic behind this program or learn how to run it. However, if you are interested in doing a test like this for your project then you can find it at http://www.phylodiversity.net/bmoore/software_symmetree.html. You should be aware that it is a command line program.

Tree simulations and tree balance statistics in R

Please download and run:

ib200b_lab11_tree_shape_clade_imbalance_v1.R

…and answer the questions at the end in email.

Please make the subject of your email:

“IB200B lab11 by [your name]"

e.g.:
Subject: IB200B lab11 by Nick Matzke