Lab 1: Introduction to *Mesquite*

Today we will have a brief introduction to some of the features of the *Mesquite*, and learn about the Nexus file format, which allows *Mesquite* to be integrated with many other phylogenetics programs. *Mesquite* was developed by Wayne and David Maddison (twin brothers, both in phylogenetics; not joking) as a tool for interpreting phylogenetic information. It is available for Mac, Windows and Unix at no charge from http://mesquiteproject.org/mesquite/mesquite.html. Note that *Mesquite* is not designed for searching for cladograms, although it does have some very limited search capabilities. Instead, it can be used for creating and editing data matrices, examining the distribution of features on a phylogeny, and testing hypotheses about character evolution. *Mesquite* is a great way to explore the data you collect for your term paper project. However, for finding the "best" phylogeny (ie: maximum parsimony, maximum likelihood or whatever criterion you chose) you need to use another program such as *PAUP** or *TNT* (parsimony programs), *RAxML* or *GARLi* (likelihood programs) or *MrBayes* or *BEAST* (Bayesian programs).

Mesquite is modular, which means that the program is set up as a bunch of modules that all do different functions, such as draw a picture or do a parsimony analysis. Some modules use other modules and the modules are used in combination to perform an analysis. Thus *Mesquite* is very flexible and capable of doing analyses that it was never intended to do. Unfortunately it also means that *Mesquite* can be difficult to use, because it is not always clear where to find the appropriate command in its menus. Today we will learn some basic properties of *Mesquite*, which we will use a lot as the semester goes on.

Other things to download: free tree-viewers. Mesquite is useful for manipulating character data, but fairly clumsy (not hopeless) with trees. Sometimes, when you just want to look at a tree, these can be easier.

Dendroscope: http://www-ab.informatik.uni-tuebingen.de/software/dendroscope

Figtree: http://tree.bio.ed.ac.uk/software/figtree/

Creating and Editing a Data Matrix

Note: Enter your own project data in this exercise if possible. If you don't have any character data yet, you can make it up.

Open Mesquite; three windows will open. The command window is just black with letters. It is the platform on which *Mesquite* runs, closing it is the fastest way to quit the program. The Start Up Window will be open just as the program boots. The ***Mesquite* Log** gives a constant narration of what the program is doing. It is sometimes useful to look at this window,

and it is the basic window for controlling files, but most of the work in *Mesquite* is done elsewhere.

Select **File>New**. Choose a location for your file, where you'll be able to find it in the future and give it a name with a ".nex" file designation (eg. "Nicks_temp.nex"). A **New File Options** window will appear. Check **Make Taxa Block**, give it a name, stay with 3 taxa, check **Make Character Matrix**, and hit **OK**. In the **New Character Matrix** window make up a name, and stick with two characters. The options in the **Data Type** menu are pretty self explanatory, and we will study most of them later in the semester. For this lab we'll stick with **Standard Categorical Data**. Hit **OK**.

A Project window will open showing an empty **Character Matrix**. Here you can edit and add characters and character states. There are several tools along the left side that allow you to manipulate the matrix. When you hold the cursor over each of the buttons, a description of what it does appears at the bottom of the window.

To the far left is a panel showing some of the other viewing options. Click the **Taxa** button. The **Taxa Block** can be used to edit information about taxa. Let's just change their names. Click on the **edit tool** (the little cursor:) and use it to select each of the taxon names, and name those taxa.

Now go back to the **Character Matrix** by clicking the tab at the top of the window, or the button on the far left. Now let's add some data to the matrix. Select the **edit tool** again and click on the various cells in the matrix. Since this is a categorical data matrix, all the characters must be consecutive integers starting with 0. So fill the matrix with a bunch of 0s and 1s; if you want to go really crazy, you can even add some 2s.

Note: Tools. To the left of the data matrix are a series of tools.

Move the mouse over them, and a text description will appear <u>below</u> the character matrix. Mesquite matrix editing makes a lot more sense when you know what these tools are.

Question #1: *Below, write the tool name and a short description next to each button.*

Select the **Add Characters** tool from the menu on the left. Click on the matrix and add 3 characters. Now select the **Add Taxa** tool and add 2 more taxa. Now you want to fill in all those empty cells in the matrix, but don't just use the **edit tool**. Click on the **Copy States** tool (the eye dropper) and use it to select a state from one of your cells. Now select the **Fill with missing data** tool (the spilling paint bucket) and click on a cell to fill it with the character state you selected with the **Copy States** tool. You can fill cells with a different character state by clicking the **Copy States** tool again and selecting a new character state.

A matrix filled entirely with 0s and 1s is often not the easiest thing to interpret; we would be much better served, if the characters and character states were more descriptive. At the bottom left are for small buttons next to a blue ***i*** that look like little windows. Select the one second from the right, the **Show State Names Editor Window**. Select the **Edit** tool and use it to give names to some of the characters and their states. For example maybe character 1 is "common sense" and its states could be 0, "absent", and 1, "present". Now go back to the **Character Matrix** by clicking the tab at the top of the window. As you can see all the 0s and 1s have been replaced with actual words. One thing to note is that when adding or editing character

states in the matrix you still need to type in 0 and 1, even if the states now have more interesting names.

(add as many of your characters as you like, then continue)

We have only looked at a fraction of the tools available for editing the data matrix, and you might want to spend some time either exploring the other tools on your own, or going through the *Mesquite* tutorials.

Creating and Editing Trees

In the Project window that you have already been working on select **Taxa & Trees > New Tree Window > Edit Trees From Source.** OK, so there are a ton of options here. We will deal with many, but not all of them in future labs. For now just select **Default Trees** and hit **OK**. A new tab will appear with a tree of your five taxa in it. On the left side at the top there is a menu that allows you to scroll through three different trees that *Mesquite* created. Take a look at all three. Below that are several tools for modifying the tree. If you want, try messing around with some of these functions.

Question #2: *I know it's a little bit of busy work, but your Mesquite life will improve if you know what these tools are and what they do. Give names and short descriptions below.*

Click on the **Move Branch** tool (an arrow). Use it to grab one of the branches of the tree and drag it to another branch. That should change the topology of the tree. Do you see the difference? Move some more branches around until you get the hang of it. Make sure you move both internal and terminal branches.

Click on the **Collapse Branch** tool (three branches as a polytomy). Use it to click on an internal branch. That branch will collapse, creating a polytomy.

Now click on the **Reroot at Branch** tool (a circle with a downward pointing arrow). Use that tool to click on any branch other than the two that emerge from the root and the root will be moved to that branch. See how it works?

You can remove taxa or even whole clades from the tree with the **Prune Clade** tool (scissors). Select it and chop out a clade.

Finally there are two ways to adjust branch lengths. Here we will just use the more exact method. Select the **Adjust Branch Lengths** tool (a vertical ruler with an arrow pointing at it, not an arrow pointing vertically next to it). Click on a branch and a small dialog box will appear. Type a branch length into that box and hit enter.

That's enough for you to make any tree, but many of the other tools can make tree manipulation more efficient and allow you to do some things that will be useful for later labs.

Nexus file basics

Let's save all of our work. First we need to store the tree. I don't know why we need to do this as a separate step, but we do. Select **Tree>Store Copy of Tree as...** Give the tree a name hit OK and then give the tree block a name and hit OK. This will store the tree that you currently have up on the screen. Now save all your work **File>Save File**.

Now navigate to the file and open it in a text editor; we're going to explore a Nexus file.

Nexus is a format used by many different phylogenetics programs. By having a common file format the results of one program can be viewed or analyzed in another (although this does not always work as well as one would hope). You can manipulate a Nexus file directly in a text editor, but often it is better to use a program like *Mesquite* and save the results in a Nexus file to be used in another program.

Every Nexus file starts out with "#Nexus" and then is followed by a brief description of the file surrounded by brackets. This is followed by the actual data of the Nexus file, which are organized into several **blocks**. Each block starts with a line "BEGIN *BLOCK NAME*;" and finishes with an "END;". The lines between, which hold the actual data, are often indented. Each program creates and uses different blocks with information and commands that are particular to it, but there are several block types that are almost universally used and contain the most fundamental information for phylogenetic analysis.

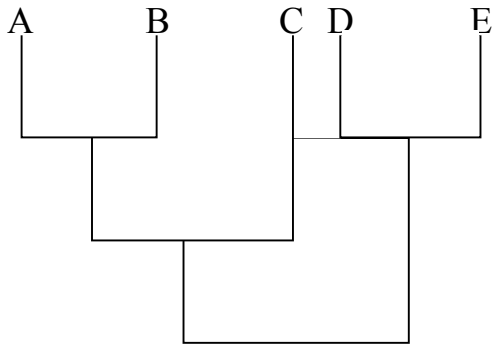
The **taxa block** is pretty much just a list of taxa. It usually contains a name for the block, a command describing the number of taxa, and a list of the **taxalabels** (in other words the names of your taxa).

The **characters block** contains your data matrix. First there is a **title** line and then a **dimensions** line, which can contain information about the number of characters and taxa. This is followed by the **format** line, which describes the data in the matrix. **Datatype** determines the basic class of data: discrete; continuous; protein; or DNA. After that there are several commands, which describe what symbols are allowed and what they represent. For example, does a "-" mean unknown or missing. After the format line is the actual **matrix**. Each line of the matrix starts with the name of the taxon represented and is followed by a series of symbols representing the character states for the various characters in the matrix. Each taxon line must have exactly the same number of characters in the same order. The characters block may also contain other information such as **charlabels** (names for the different characters) and **statelabels** (names for the different character states for each character) or **charstatelabels** (both types of info combined in one command).

Many programs use a **data block** instead of a taxa block and a character block. Most programs, but not all, are flexible and can use either format. The data block contains all the same information as the other two blocks, except it does not need taxalabels as they are described in the matrix

You may be surprised to learn that the **trees block** contains any trees used in the analysis. A nexus file will only contain a tree block, when it is necessary to import a tree into a program.

Tree blocks often start with a **translate** command, which is required for a number of programs, but not all. It is a list of consecutive numbers followed by the names of the taxa that those numbers will represent. This is followed by a **tree** line, which describes the tree in **Newick format** using the numbers assigned in the translate command for the names of the taxa. Newick format is a commonly used way of representing tree topologies as text. Put simply monophyletic clades are surrounded by parentheses and sister clades are separated by commas. Such that the tree



would be written as:

(((A,B),C),(D,E));

Newick format can also contain information about branch lengths (after colons) and node names (after closed parentheses).

Newick files vs. Nexus files

We will be working a lot with trees in this class, and therefore tree files. You can avoid headaches by knowing the difference. Note that there are three main ways trees are stored in a computer:

1. As a Newick file (see http://evolution.genetics.washington.edu/phylip/newicktree.html for what amounts to the official format description. The Newick file is just a text file containing the character-string representation of the tree, above.

   Extra credit: tell me what the origin of the name "Newick" is:

2. As a NEXUS file, containing one or more Newick strings in a trees block.

3. As an image ("flat") in a PDF, journal publication or other graphical format. These trees are nice to look at, but you can't input them into any programs. Typically, to get these

trees as Newick or NEXUS format, you have to do one of (a) get it from the Supplemental Material; (b) email the authors; (c) download it from e.g. Treebase; (d) get the raw sequence/character data and make the tree yourself; (e) digitize the tree.

A program that used to be used for (e) was called "TreeThief", but it appears to be no longer maintained. TreeSnatcher is supposed to be a replacement, but I couldn't get it to work on my Mac. I have hacked together a replacement using R and a cheap digitizer called GraphClick. Google "TreeRogue" if you want to explore this.

The point: Some programs operate more/better with NEXUS files (e.g. Mesquite), some with Newick files (Dendroscope, Figtree, although I believe they can take either format). Sooner or later you will encounter headaches about formats, Newick strings that can't be read because of a missing ";" or ",", etc. These are a headache.

Remember that you can always open any of these formats in a text editor, copy out the Newick string, and manipulate/fix it that way.

Question #3. Save your tree in Newick format, open the tree up in Dendroscope or FigTree, and print out an image of it (screen-capture, PDF, whatever).

Commands to get a Newick tree out of Mesquite:

Tree → Store Copy
Export → Phylip (tree) *(Note: for some reason, probably ancient history, Mesquite doesn't call it "Newick" format, it calls it "Phylip" format)*

Notes:
* on Macs, Command-Shift-4 lets you send an arbitrary chunk of the screen to a PNG image file on the Desktop;
* on Windows, it should be ALT+PRINTSCRN for whatever window is active (copies the image to the Copy/Paste buffer I think, so paste into e.g. Word or an email). PrintScrn captures the whole screen.

As I mentioned before, there are several other blocks that contain information for specific programs, such as **Assumption blocks, MacClade blocks, Mesquite blocks, PAUP blocks, and Bayes blocks**. *Mesquite is* capable of creating these blocks for several other programs.

I have not given you a detailed description of the Nexus format. If you want to change a Nexus file by hand, I would recommend that you use another Nexus file as a template.

Question #4: Print out for me your NEXUS file.

Some Other Cool Features

Viewing Multiple Windows at once

You do not have to look at each tab separately it is often useful to look at two different windows at once. To do this click on the little swishy arrow on the Tree Window tab. Select either option depending on your viewing preference.

Viewing Results as Text

All these pictures are very nice, but often you want a more utilitarian output. Every window in *Mesquite* has a text tab at the top of it which displays the same information that you see in graphic form as text. Hit the **Text** tab at the top of this **Tree Window**. The text tab is particularly good for likelihood, because you can see the reconstructions as a table. Another useful feature is that you get the tree in Newick format. Hit the **Graphics** tab again when you're done looking at figures today.

Discrete Character State Reconstruction Using Parsimony

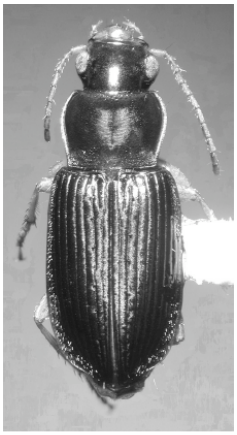


Fig. 1.—Habitus of *Amblygnathus georgei*, new species. SBL = 5.48 mm.

In *Mesquite* open the **Amblygnathus** file that you downloaded from the web site. This is a data set from the *MacClade* examples containing characters for a group of North American ground beetles.

(Just so you can see what they look like, see at right an image from: Shpeley, D. (2008). "A new species and new records of *Amblygnathus* Dejean (Coleoptera: Carabidae: Harpalini) from South America." *Annals of Carnegie Museum* **77**(1): 21-24. http://dx.doi.org/10.2992/0097-4463-77.1.21)

The whole purpose of *Mesquite* is to analyze how characters change on trees. There are two very broad categories of data: discrete and continuous. For the next couple of labs we will deal with discrete data and move on to continuous data next Thursday. There are also several ways to reconstruct ancestral character states for discrete data. We'll start with good old familiar parsimony.

A new project window will open. Click on the **Character Matrix** to see what this looks like for this data set. Now click on the trees to open the Tree Window. From the **Tree Window** go to **Analysis > Trace Character History > Stored Characters**. Select **Parsimony Ancestral States** and hit **OK**.

The ancestral states will appear on the branches. You can go through the different characters by clicking on the arrows in your **Trace Characters** box at the bottom right of your **Tree Window**. Pick an example with an ambiguous reconstruction.

Question #5. What character did you pick with an ambiguous reconstruction?

In *Mesquite* all the branches are shown with the colors of all the possible states that they could have under parsimony. When you move the cursor over a branch, you will see the character state at the bottom of the **Trace Character** box. On my computer I had to move this box up to see its bottom.

When you change a character state in your matrix the results will automatically appear in your tree. Go to the **Character Matrix** window and change the character state of the character you are currently tracing in your tree window for one of the taxa. The character state will automatically change in the **tree window**. This is a general feature of *Mesquite*, whenever you change something in one window it will effect what is happening in all the windows that are related to it.

Making Pretty Trees

You can change the colors of the character states by double clicking on their colored square in the **Trace Characters** box. Select a color and the tree will change. *Mesquite* tends to use black and white as a default but it is often easier to visualize if you use different colors.

Let's look at different shapes of trees. Go to **Drawing > Tree form >square tree**. The shape of the tree will change. Try some of the other shapes. You should do the rest of the lab in **Balls and sticks**, because it allows you to see equivocal states more clearly. There are many other options under the **Drawing** menu that allow you to change the appearance of your tree. Try changing the **line size** and **line style** to make your tree more appealing.

Changing Parsimony Models

It is also possible to change the parsimony model under which a particular analysis is run. Go to **character 5**, which has several character states. Currently the model is set to **ordered**. Select **trace>parsimony model>unordered**. This changes the model for this particular character. Does it effect the character state reconstruction? You might notice that there are two other models: Dollo and irreversible. Unfortunately they don't work yet, but they are there so that you can set the model for *MacClade* or *PAUP**. Hopefully, in the near future, they will work in *Mesquite* too.

You can also construct your own parsimony model by building a step matrix. Go to **Characters>New Character Model>Stepmatrix**. Make up a name for the model and hit **OK.** Now you need to set the maximum state by selecting **Step matrix>Set maximum state...** The number of your maximum state depends on your character. Since you start numbering states at 0 the maximum sate will be one less than the total number of states for a character. For example, if you're still on Character 5, your maximum state should be three**.** Now click on the individual cells with the text select tool to make changes to your step matrix. When you go back to the tree window you should find your step matrix under the list of parsimony models (**trace>parsimony model>**). Select it and look at the results.

Trace All Characters

It is also possible to create a table showing the character state reconstructions for each character on each branch. Select **Analysis >Trace all characters**. Then select **Parsimony Ancestral States** and hit **OK**. A **Trace All Characters** window will appear showing the reconstruction of character states for every node that you have selected. To select specific nodes hold down control and left click on them.

Discrete Character State Reconstruction Using Likelihood

It is also possible to reconstruct ancestral states for discrete characters using likelihood. Under this model there is some stochastic rate at which one character state changes into the other. The probability that a character changes from one state to another along a branch depends on the branch length and the rate of change. Thus you can assign a probability that any particular state occurs at a given node, whereas in parsimony you can only say that it may or may not occur there.

If you had not already set up a trace, then it would be necessary to first set up a new trace from the **Analysis** menu, but, since you already traced a character using parsimony, you can just change the reconstruction method. Go to **Trace>Reconstruction Method>Likelihood**. Select **Current Probability Models** and hit **OK**.

The reconstructions of the internal nodes will now be filled in such a way that it is proportional to the probability of each state at that node. If you hold the cursor over a node the **Trace Characters** box will show the probability of each state at that node.

Probability Models

In likelihood models the probability of changing depends on the rate of change. How do you determine what the rate of change is? The default in *Mesquite* is to assume that the rate of change is constant on the tree. That value is itself determined empirically through maximum likelihood using the distribution of character states found on the tree. The empirically derived rate and the log likelihood can be found in the text in the middle of the **Trace Characters** box.

Question 6. What is the estimated rate for character #7?

However, there are other possible models to implement in *Mesquite*. For example you can have one rate for changes from state 0 to state 1 and another for changes from state 1 to state 0. The default is to have the two rates the same and this is called **MK1**. If you allow the two rates to vary independently it is called **Asymmetric**. For characters with more than two states *Mesquite* can only calculate a one rate model, however *R* can make the calculation with any set of rates between states for a multistate character. I don't think we will cover this in class, but if you're interested ask me about it after we get into *R*. To use an asymmetric model in *Mesquite* go to a two-state character and select **Trace>Probability Model> Asym. 2 param. (estimate)**. The estimates of probabilities at the nodes will change and you can find the estimates for both rates in the **Trace Characters** box.

It is also possible to set the rate of change yourself, rather than determining it empirically. Select **Characters>New Character Model > MK1 model**. Name the new model and hit **OK**. You can change the rate parameter by sliding the bar or typing in a new value and hitting the arrow next to the rate, and the results will be displayed instantly on the tree. To implement your new model select it from **Trace>Probability Model>**. It is also possible to implement an asymmetric model where you set the parameters yourself.

Branch Lengths

The other important parameter in a likelihood analysis is the branch lengths. To view the branch lengths of a tree go to **Drawing>Branches Proportional to Lengths.** (It would definitely help to have your **Line Style** set to **Square** here.) As you can see all the branch lengths are the same, like in a parsimony analysis. Since the tree we used had no branch lengths saved with it, *Mesquite* assumed that they were all the same. In reality you could get branch lengths from a number of different sources, such as a likelihood analysis of a larger data set or estimations of the actual node times from the fossil record.

There are two tools to alter the branch lengths according to your whim. The vertical ruler with an arrow pointing at it (fourth from the bottom on the left) allows you to type in a new branch length by clicking on the branch. The vertical ruler with two arrows pointing up and down (fourth from the bottom on the right) allows you to change branch lengths by dragging the node. Change the lengths of some branches and see how the reconstructions change. The ones with ambiguous reconstructions will give you the most interesting results. Does extending the length of a branch make the nodes on either side more or less like each other?

It is also possible to set the branch lengths automatically. Go to **Tree>Alter/Transform Branch Lengths > Ultrametricize.** This will make it so that all the terminal nodes end at the same time, as they would under a constant rate of change, like the molecular clock. There are several other options for automatically setting branch lengths.