GENOME 453 Autumn, 2015
Evolutionary Genetics J. Felsenstein

Phylogeny computer program exercise

Due Wednesday, 11/25 (this takes some work so don't wait until the last minute)

This is an exercise in which you will each analyze a data set by finding the most parsimonious tree (that you can), using a program, `Dnamove`, from my phylogeny package PHYLIP. Each of you will be using a 14-species data set, each one of you with data from a different locus. Then, after you turn in the tree to me, we will compare them and compare them with a tree that comes from all the different loci.

The data come from the OrthoMam database of aligned sequences from mammals. I chose the loci from those for which these 14 species were sequenced.

Do the following:

1. **Get your data set**. Go to the downloads web site for this exercise:
   `http://evolution.gs.washington.edu/gs453/2015/downloads`
   There you will find your data set. Take the last two (or three) digits of your UW student number and add it to `infile.`, so if your student number is (say) 10658236, your data set will be `infile.36`

   In two cases (numbers 65 and 78) two students have the same two last digits, so they should use their last 3 digits.

   You can save the data set by using the `File` menu of your web browser, and its `Save As` choice.

   In addition to saving it, make a copy of that file called `infile`, as that is the file name the program will be looking for.

2. **Get the program**. It is in that same folder. There are three forms of it:

`dnamove`	A Linux executable (for Intel-compatible processors)
`dnamove.exe`	A Windows executable
`dnamove.macosx.gz`	A Mac OS X executable (for Intel Macs)

   You can download and save these files – the Mac OS X file is a Gnu-zipped archive that should be extracted and saved. If your operating system is unhappy about downloading a zipped file, or a file from the internet, you may need to confirm that you want to do that. For recent versions of the Mac OS X operating system, if you click on the icon of the Dnamove program, Mac OS X may not allow the file to open. Instead control-click on it, and select Open in the menu that appears. Then verify in the box that appears that yes, you do want to open the file. (It is from an unverified developer – me, because I neglected to sign that file with my Apple Developer ID).

   PHYLIP programs such as Dnamove are also, alas, not yet available in versions that work on tablets (Apple iPad or Android tablets) so you will need to use a different computer in that case. If you manage to make PHYLIP work on a tablet, please let me know how you did it.

Aside from the description in this assignment, if you need more documentation on how to run Dnamove, or about PHYLIP more generally, documentation will be found as web pages at these links:
`http://evolution.gs.washington.edu/phylip/doc/dnamove.html`
`http://evolution.gs.washington.edu/phylip/doc/main.html`

In some cases you may have to open a window that has a command-line and make sure you are in the right folder, then type the program name. On Windows the tool to use for this is called Command Prompt and is found in the All Programs list, under Accessories. For Mac OS X the tool is Terminal which will be found in the Utilities folder inside the Applications folder.

If you can't make one of these work, you can also go to my PHYLIP web site (guess what you type into your search engine to find it?) and download and install a copy of the whole package. Dnamove will be in there.

3. **Run the program**. Make sure that `infile` is in the same folder as the program. If you have a Windows system or a Mac OS X system, click on the Dnamove program icon. If you have a Linux system, type the program name `dnamove` or (depending on how your command path is set up), type `./dnamove`

   A window will open and you will communicate with the program by typing responses into the window. If the program says that it can't find the input file `infile`, and asks for you to type in the file name, try some alternatives like `infile.` or `infile.txt`, as sometimes the file name is one of those when you think it is `infile`. Or you can type the actual file name that has the digits from your UW student number. If the program still cannot find the file, make sure that the file is in the correct folder, the one that you are running the program from.

   Note that if you happened to read the file into a word processor such as Microsoft Word, and then saved it, it could have been converted into a Word format document, instead of a simple Text format. The Word format has all sorts of horrible stuff in it and will cause PHYLIP programs to become totally confused. If you save the file, you want to make sure that it is saved in Text Only format/

   I will give some instructions below, but for more information on options of the program, people can look at the documentation web page `dnamove.html` which will be found in the same `downloads` folder that has the executables.

   The program presents a menu for some run settings. When you have done doing those (the first time, just accept the defaults), answer Y to accept the settings. It then creates a tree (arbitrarily) and then asks you what you want to do with it by presenting a single-line menu of command characters below the tree. If you can make the window large enough and make the right setting for L, the number of Lines in the window, you may be able to see the whole tree at once.

4. **Rearrange the tree to find the most parsimonious tree.** First use the `O` (Outgroup) command to make species 12 (Ornithorhynchus), the duckbill platypus, the outgroup, because we are pretty sure it is. That makes the tree sensibly rooted.

   To use parsimony to reconstruct the best tree, take advantage of the ability to rearrange the tree. One good way to do this is by using the `T` ("Try rearrangements") command in the tree rearrangement menu.

   After you type `T` and the number of the node (say it's 3), the program will show you possible placements of that group, together with the number of steps (changes of state) that each requires. You will want to search through the `BETTER:` list and find one of the ones that has the lowest number of steps. The group has been put back in its original location once it was tried in all possible places.

   You're going to be asked to run the `T` command on species 1, then on species 2, and so on, including doing it on the interior nodes of the tree. All these nodes are numbered. The interior node numbers follow the tip nodes that are numbered one after another (1 to 27). You will not be allowed to do the T command on the rootmost interior node. Make sure that in addition to Trying new locations for the tips (nodes number 1 – 14), you should also try new locations for the interior nodes.

   If a `BETTER:` location was found, now you want to move the group (or species) to that new location. Use the R ("Rearrange") command to move it to that node. For the number of the node to remove, use the node number that you used in the T command. For the number of the destination, use the number of the node that was in the `BETTER:` list. Make sure you use the node number, the number before the colon, as the destination, not the number of steps, which is after the colon. Choose the `B` (Before) setting, rather than the A (At) setting when it asks you about that.

   Keep doing this on all the nodes in turn, until none show any better placements. If part of the tree scrolls off the screen, the dot (.) command can make it reappear. If only part of the tree shows, you can use the `H`, `J`, `K`, and `L` commands to see that part. Note the report up at the top of the screen on how many steps (changes) are now needed, and whether or not this is the best number yet. Don't concern yourself with the number about how many characters are compatible with the tree.

   When the tree cannot be improved, use the `Q` command to quit the program, making sure you also choose to write the tree out (it will go into a file named `outtree`). If it asks you whether you are sure you want to overwrite an existing file of that name, think about that – if you wanted to save that one, I hope you already made a copy of it under another name. If you didn't save that one, the program allows you to chose a different file name for the tree you are writing out. You might want to make sure there is a copy of that final tree file with some other name, so it doesn't get overwritten.

   The file you ended up with is visible on the Dnamove screen. It is also in file "outtree" in a computer-readable format called the Newick format. While you can find web pages on the "Newick tree file", you can also view the tree using the PHYLIP program Retree. But if you don't do that, you could also run Dnamove, this time telling it to read the tree from the user-defined tree file (it will assume the name is "intree" so you would need to first make a copy called that). Then you will see it again. Make your window big enough, and the number of Lines (option L in the Dnamove window) big enough, and you should see the whole tree.

I repeat, make sure you make a copy of your final tree file – you need to have it to send it to me.

Do *not* end up reporting to me the tree that first appears on the screen when you run `Dnamove` – it is a completely arbitrary tree that is not a good estimate of anything. You have to do rearrangements to improve it.

Note also that the tree's parsimony score does not depend on where we chose to put the outgroup, it is just that this makes it easier for us to see what monophyletic groups are in the tree.

By the way – if you want to interrupt a run and resume it later, you can write out the tree to a file `outtree` and then make a copy of it called `intree`, and when you run the program again, you can use the initial program menu and have it read that tree in as the initial tree.

5. **Reporting the result.** Send me a brief email with the tree you found, and the number of steps (base changes) that it required. *Important:* attach to that email a file with the tree (the nested-parenthesis version that the program produces and which it put into a file called `outtree`. In your report, also comment on what aspects of your tree seem to you to make sense. There is a file `species.txt` available at the downloads web site that tells you what kind of mammal each species is.

   Also tell me how many steps (changes) were required per site. The number of sites in your alignment is available on the first line of the data file. Divide the total number of steps by that number. The bigger that number is, the more rapidly that locus was changing.

   Look at the last line of the data file too. There you will find the HUGO (HUman Genome Organization) symbol for your locus, and its name. Look in the `OrthoMam` database for it (start by typing `OrthoMam` into a search engine) and see what you can find out about its function. Is your locus likely to be an essential, highly conserved one carrying out a critical cell function, or a more rapidly-changing one? Tell me what you think.

   I will be taking all these trees and making a consensus tree, and also comparing it to an overall tree inferred from a concatenated alignment. That alignment is available at the downloads website as `infile.all`. You might want to try inferring the tree from it yourself, just for fun. The tree of mammalian groups is a hard one to infer, as the adaptive radiation of placental mammals was rather rapid and was a long time ago, a bad combination. But we will have a lot of data.