# Changing tip labels using morTipLabels

## Andrew Hipp

## 2021-01-28

The function `morTipLabels` is a pretty rudimentary method for relabeling tips for our own lab data, mainly the trees generated from RAD-seq data... but it's generalizable. To run it, first make sure you have installed `ape`, `magrittr`, and `openxlsx`. Then, you'll need to grab the function off of GitHub:

```
source('https://raw.githubusercontent.com/andrew-hipp/mor-systematics/master/AAA.LAB/morTipL
```

This reads the function into `R`. It also assumes you might want to use some of the specimen metadata files from the T: drive, so it *tries* to read those in; it should work fine even if it doesn't succeed. To see whether it has succeeded, look at the contents of your workspace:

```
ls()
```

```
## [1] "a"            "dat.acer"     "dat.malus"    "dat.tilia"    "morTipLabels" "tidyName'
```

You should see `dat.acer`, `dat.malus`, and `dat.tilia` in your workspace. If you do, then hooray! If not, then the T: drive was probably not mounted at the time that you sourced in the file.

You'll then need to read your tree in so that it is in `phylo` format. To do this, use the `read.tree` function from ape. If you are using RAxML output, I recommend getting the "bipartitions" tree; it seems to read well and has the bootstraps in there so you can plot them if you want.

```
tr <- read.tree('https://raw.githubusercontent.com/andrew-hipp/mor-systematics/master/AAA.DA
```

Then, you call the function, like so:

```
a=morTipLabels(tr, dat.acer, outgroup = 'saccharum', outfile='RAxML_trTest')
```

```
## Rooting by this outgroup:
```

```
## Acer saccharum|Illinois|USA|474-74*2|ACER-MOR-194|ACERspm000209
```

Of course, that's just a simple example, so let's look at what your options are. The arguments of the function are defined below; all with an "=" sign:

**treesToDo** A single tree or a list of trees, all read in using `read.tree` from ape

**dat.meta** This is a *specimen* database from our lab. While you don't actually have to limit yourself to the specimen database, it makes more sense for us to keep those live and always do tip label conversions from them directly, using the active T-drive version.

labelCols = c("TAXA-Current_determination", "State", "Country", "Collector_no", "RAD-SEQ-FLORAGENEX", "SPMCODE") : These are the columns of `dat.meta` that you want to be in your new tips, in the order in which you want them. They don't have to match up completely, so long as you set `grepCols = TRUE` (which is the default). More on that below.

**matchCol = "RAD-SEQ-FLORAGENEX"** This is the column that has the file name matching your tip labels. Defaults to the RAD-seq Floragenex name, b/c we're so into RAD-seq trees... still...

**grepCols = TRUE** If `TRUE`, then the column names above are matched using `grep`, which will select any string that *contains* the name. The reason I am using `grep` here is that the databases don't all match up. When there are multiple entries that grep can match to a name (e.g., "Country" matches to "Country" or "GeoReferenced.Country"), the function only keeps the first one.

**tipRemovals = ".fq.gzbarcodeStripped"** Have text you want to strip off your tips b/f you relabel them? put it here. If there are different texts you want to remove, pipe delimit them with no spaces, e.g., `tipRemovals = ".fq.gzbarcodeStripped|extraData|revAndComp"`. All will be removed.

**outgroup = NA** If != NA, sets the outgroup by which to root. This is matched *after* renaming, so you should be thinking of the relabelled tip names when you set the outgroup.

**outgroupGrep = TRUE** `outgroup`

delim = "|" ladder = TRUE, outfile = NA pdfTree = TRUE tip.cex = 0.6 nodeLabels = TRUE pdfW = 10 pdfH = 20