Using Y-Utility to Sort Haplotypes Into Patrilineage Groups 1

The online program, Y-Utility (YU), authored by Dean McGee, provides a comprehensive method of classifying DNA Project haplotypes into patrilineage groups—tested males, usually with the same surname, who all descend from a common ancestor who lived since the time the surname was adopted in that line.

The technique, basically, is to copy-and-paste the DNA project data from its display matrix into a TXT file (though it could be done with a wordprocessor too) clean it up a bit, then feed it into YU to produce a Genetic Distance matrix, showing the closeness of relationship between each pair of project members. I have outlined the procedure I follow below in full detail.

Gather the Data From the Project Haplotype Tables

Cut and paste the haplotypes, and their identifiers, for all the project groups to be analyzed.

For convenience, this data should be processed in batches, one for each major haplogroup, e.g. Haplogroup R1b, I, E, etc. The major haplogroup for each testee is estimated by FTDNA from his haplotype results, and a person's more articulated haplogroup can be determined through additional, "deep clade" testing. Once a group has been constructed, a single member's deep clade test will apply to all the other members, but for purposes of constructing groups in the first place only the broadest haplogroup classification should be used.

Prepare the Pasted Haplotype Data for Input to YU

Weed out all haplotypes that are not FTDNA-37 or -67 and truncate the latter to -37. Each 37-marker line should end with a pair of marker values in the 30s (occasionally one of these values will be 40), followed by a pair in the low teens.

Append a group identifier to the project member identifier for each remaining entry, e.g. "R1-" to identify members of Haplogroup R1b Group 1.

Delete all text between the project member identifier and the first marker value, leaving exactly one space in front of the latter.

Align the group-member identifiers for each haplogroup, if necessary padding shorter ones to the right with decimal points. Align all marker values with exactly one space between each. If necessary for alignment, prefix single digit marker values with one zero.

Each set of markers for each group and each haplogroup should be aligned. Alignment isn't necessary for YU, but keeping the data aligned allows one to see whether it's complete just by eyeballing the group of haplotypes.

What *is* necessary for YU is to ensure that there's exactly one space between the identifying header (in this case the GroupPrefix-MemberID) and the first marker value, and between each of the other marker values.

The aligned data should look like something like this:

```
R1-108 13 23 14 10 11 14 12 12 12 14 13 30 17 9 10 11 11 25 15 18 32 15 15 16 16 11 11 19 22 16 14 18 17 38 38 12 12
R1-109 13 23 14 10 11 14 12 12 12 14 13 30 17 9 10 11 11 25 15 18 32 15 15 16 16 11 11 19 22 16 14 18 17 38 38 12 12
```

where "R1-" means "Haplogroup R1b #1" or whatever unique group you want it to designate, and "108" is the unique identifier of one project member (this might also be a Kit#).

When the data is formatted and appears to be correct, save it to a permanent YU Haplotype input file so that you don't have to do the above again (except for adding new members who test or extend their tests to 37 or more markers).

From this point on, the way you group and feed the data into YU depends on what you're trying to accomplish. Whatever you do though, make sure that the original formatted data is left intact and never changed once it's formatted. The best way to do this is to make a copy of it (or of part of it) and do your regrouping for YU with the copy, then, once you're through with the copy, discard it so that you only have to maintain one original set of data.

Using YU to Add New Members or to Verify that the Haplotypes of an Established Project Are Correctly Sorted into Patrilineage Groups

If you have a large project with many groups and many singletons, and you want to add a new tested member to the appropriate group, or simply to make sure that the present grouping is correct and that none of the singletons actually belong in one of the groups, select all the formatted, aligned haplotypes that fall into the same haplogroup, both patrilineage clusters and singletons (or if you have some very large patrilineage clusters, select one representative member of the cluster). Input this data into YU to produce a genetic distance chart and look for values in the cells that are in the single digit range. Any colored cells that turn up mean that the corresponding row and column items belong in the same group. Uncolored GD numbers from 6-7 probably belong in the same group, while 8s and 9s are more problematic and require further analysis.

To resolve ambiguous matches, singletons that match to an established group member at GD 6-9 should be separately combined with the whole group from which that member was selected and run through YU with that group. If any colored cells turn up for the questionable additions, then these qualify as members and should be amalgamated with the group.

Otherwise (where the GD numbers are still in the 6-9 range), the decision of whether to include the singleton depends on the frequency of the surname (high value GDs of 8 or 9 for ultra-common surnames like Smith or Walker should probably not be combined) and on whether the research for the singleton has turned up circumstances in common with any of the more closely matching group members.

For example, in my Robb patrilineage, several Robb families immigrated, probably from Ireland or Scotland, to the western counties of Pennsylvania during a span of 40 years and lived within a radius of 30 miles. Under these circumstances, and given that Robb is a less than common surname, GDs of up to 10 might be considered close enough for inclusion in the group.

If the singleton still fails to qualify, it should be left in the singleton pool. It may be that as new members join the project one of them will turn out to be a "tweener", having a haplotype intermediate between one of these group outliers and the core of the group. Subsequent YU runs may thus link these outliers closer to the existing group members and finger them for inclusion at that time.

Running YU to Produce a Genetic Distance Chart

Go to YUtility.

Uncheck all the parameters in the left column except the "Genetic Distance" report; then check the "Hybrid mutation model" sub-parameter. This will ensure that multiple marker differences will be counted as separate mutations, except for markers DYS464 and YCA.

Under the "General Setup" parameter block, uncheck "Create Model Haplotype".

Under "Highlight Reference", check "None".

Concatenate all your data together into a single block of aligned haplotypes, each with its Group-Member header, and copy and paste this block into the YU input box

Click "Execute" and the Genetic Distance chart will be generated in a separate window.

YU does a range check on each of the marker values so that if any are missing or out of the known range of allele values, you will get an error message instead of the chart, and the error message should steer you toward the source of the problem. YU is picky about its input, which is a good thing because it means that if it runs at all, you probably have meaningful data (no "garbage in, garbage out" allowed).

To save a copy of the GD chart, you will probably have to do a screen snapshot. If you don't have a utility program to do this, you can find dozens of them at the CNET download site . I use one of the more popular free ones, called ScreenPrint32, which can be downloaded here (by all means read the CNET reviews first so that you have some idea of the pros and cons of what you are getting. The way this utility works (if you set up its parameters the way I have) is that you hit the "PrintScreen" key, which freezes the screen, then you select the rectangle you want to capture (make a savable picture of) in the usual way by clicking and holding the left mouse button while you size the desired rectangle by dragging, then right click to actually save it (or to cancel the save if you fluffed the drag 'n drop). Where the images get saved is also a definable user parameter. I save my images to a high-level directory named \Screen so I can later access it quickly and easily—don't let your screenshots get buried in some obscure C: subdirectory. You can also set up ScreenPrint32 to save images in several formats, including JPGs. I use GIF format because it's sometimes more compact even than JPG, and because JPG image quality can degrade if you manipulate the images too much.