ABSTRACT

Summary: We implement a program that incorporates polymorphic sites data, haplotype frequency arrays, and other factors, into cladogram estimation.

Availability: It is available at http://sdmc.krdl.org.sg:8080/~lxzhang/cladogramer

Contact: lxzhang@krdl.org.sg

Cladistic analysis is an important method for locating functional mutations or a marker that is in linkage disequilibrium with a functional mutation (Templeton et al., 1987, 2000). The idea of this method is that any functional mutation would be embedded in an evolutionary framework of prior mutations. Using this method, Hallman et al. (1994) studied the effect of variation in the apolipoprotein B (Apo B) gene on lipid levels in French nuclear families; Heng et al. (1999) investigated the association of the Apo B gene with coronary artery disease in Singaporean Chinese and Indians.

Obviously, the power of the cladistic analysis depends upon the accuracy of cladograms used to draw inferences. Haplotype cladograms are affected by several factors such as haplotype frequencies, patterns of gene flows, etc. (Templeton et al., 1987), which are generally ignored in ‘interspecific’ phylogenetic analysis (Felsenstein, 1995; Swoford, 1993). Here, we implement a tool named Cladogramer that incorporates haplotype frequency into cladogram analysis using the parsimony criterion. We adopt the following hypotheses that were predicted by coalescent theory (Crandall and Templeton, 1993): (1) the rank of alleles by age is similar to their rank by frequency in decreasing order; (2) the probability of a haplotype being derived from a more frequent haplotype is higher than one with a lower frequency. These hypotheses imply that common haplotypes usually occur in the interior of cladogram, whereas rare haplotypes usually at terminal ones.

To incorporate haplotype frequency information into cladogram estimation, we treat the haplotype frequency as a new character along with sequence-position characters, but we only allow changes from high frequency to low frequency; hence, a penalty \(w > 0\) is imposed for each change from low frequency to high frequency (called frequency-reverse change). Here, the penalty \(w\) is introduced for distinguishing frequency characters from sequence-position characters; and its value is determined in terms of individual cladogram analysis itself. Since finding cladograms with the minimum parsimony score is NP-hard, there is unlikely an efficient polynomial-time algorithm for it and hence we implement a heuristic method. We first sort the haplotype frequency array in decreasing order, and then add haplotypes to intermediate cladograms by their rank in the sorted frequency array. The higher its frequency is, the earlier the haplotype is added. The parsimony score is non-decreasing as additional haplotypes are added to a tree; hence, we use the branch-and-bound technique to speed up the program.

Finally, since rooting haplotype cladograms is extremely difficult due to the high similarity within species, and often controversial (Crandall and Templeton, 1993), we allow its users to specify a root in the input file to make Cladogramer more flexible. More specifically, an input file contains the coding (0–1 or DNA) sequences of the haplotypes being studied. The first line must contain the number of haplotypes, the number of loci; it may also contain additional parameters ‘F’ and ‘R’. If ‘F’ appears, the second line in the file must contain the list of frequencies of haplotypes. Similarly, if ‘R’ appears on the first line, the root of the evolutionary pathway must be provided on the third line. See the right-bottom frame in Figure 1 for illustration.

To make the tool easy to use, we use a web interface; we created this interface using CGI programs in Perl and C. The interface window is partitioned into three frames as illustrated in Figure 1. The input appears in the right-bottom frame and the output list of trees in left and right-top frames. Each output tree in the list has the minimal parsimony score and contains fewer frequency-reverse changes than specified as a parameter in the input frame. The user can view the results while...
keeping the input unchanged. This allows one to modify parameter values until one is satisfied with the analysis. Since multiple output trees are listed in the left-frame and individual trees are displayed in the right-top frame, one can easily examine trees one by one. The right-top frame also displays the on-line documentations of both program and parameters.

ACKNOWLEDGEMENTS
The authors would like to thank the reviewers for their many helpful comments on this note.

REFERENCES