TECHNIQUES FOR ASSESSING PHYLOGENETIC BRANCH SUPPORT: A PERFORMANCE STUDY

DEREK RUTHS LUAY NAKHLEH

Department of Computer Science, Rice University, Houston, Texas 77005, USA.
{druths,nakhleh}@cs.rice.edu

The inference of evolutionary relationships is usually aided by a reconstruction method which is expected to produce a reasonably accurate estimation of the true evolutionary history. However, various factors are known to impede the reconstruction process and result in inaccurate estimates of the true evolutionary relationships. Detecting and removing errors (wrong branches) from tree estimates bear great significance on the results of phylogenetic analyses. Methods have been devised for assessing the support of (or confidence in) phylogenetic tree branches, which is one way of quantifying inaccuracies in trees. In this paper, we study, via simulations, the performance of the most commonly used methods for assessing branch support: bootstrap of maximum likelihood and maximum parsimony trees, consensus of maximum parsimony trees, and consensus of Bayesian inference trees. Under the conditions of our experiments, our findings indicate that the actual amount of change along a branch does not have strong impact on the support of that branch. Further, we find that bootstrap and Bayesian estimates are generally comparable to each other, and superior to a consensus of maximum parsimony trees. In our opinion, the most significant finding of all is that there is no threshold value for any of the methods that would allow for the elimination of wrong branches while maintaining all correct ones—there are always weakly supported true positive branches.

1. Introduction

The accuracy and validity of most comparative genomic studies rely on the quality of an underlying "guiding" phylogenetic tree. Such a tree is often inferred using a phylogeny reconstruction method. However, such methods are bound to make errors in the inferred tree (by inferring wrong branches and missing correct ones) due to a host of reasons such as biological processes that may not be modeled by a single tree (e.g., recombination and horizontal gene transfer) or "data issues" (e.g., incomplete taxon sampling, insufficient data, wrong assumptions). Various methods have been introduced for estimating the support of (or confidence in) tree branches; two of the most commonly used methods are the *bootstrap* method[19] and Bayesian inference techniques. The bootstrap method is usually coupled with the maximum parsimony (MP) or maximum likelihood (ML) heuristic searches, and amounts to estimating many trees over subsamples of the dataset and using the the percent of trees containing a branch to be its support. Bayesian inference uses statistical inference techniques whose final outcome is a set of trees, each coupled with probabilities associated with its branches to reflect their support. Further, MP heuristics often compute a large set of optimal trees. The number of trees in which a given branch appears can be taken to be its support (these are referred to as the "consensus" methods). After support values are computed, a threshold is chosen and branches with support lower than that threshold are contracted. The hope is that a threshold exists such that erroneous branches will be removed while correct ones will be retained.

Existing simulation-based performance studies of branch support measures have considered Maximum Likelihood with bootstrap and Bayesian Inference,[5,6] as well as the

statistical properties of the bootstrap.[2,3,4] Other studies considered the performance of the various branch-support estimation methods on biological datasets, in which case the true phylogeny is usually unknown.[12,1,18] One exception is the work of Taylor *et al.* in which they studied the accuracy of the bootstrap and Bayesian approaches in reconstructing the phylogeny of several strains of yeast.[23] The results focused on studying the effect of evolutionary rate consistency and tree shape on accuracy.

In this paper, we evaluate both the absolute and relative correctness of each method under the conditions of the study by evaluating the performance of branch support assessment methods via simulations. We generate random phylogenetic trees, and simulate the evolution of DNA sequences down these trees. We study the accuracy of the trees and the support of their branches, as calculated by the methods, by comparing their estimates to the true (known) phylogenetic trees. In this study, we focused on the performance of the most prevalent branch support-labeling algorithms. Thus, we have omitted the less common distance-based branch support assessment methods, such as bootstrap with *neighbor joining*. We also do not explicitly consider the error resulting from the fact that the heuristics we consider do not actually converge to the true tree under all conditions.[20,14,22]

We focus on three main questions. (1) Is the support (as calculated by each of the methods) of a correct branch significantly higher than that of a wrong branch? (2) Is there a clear threshold for each of the methods that would allow for contracting wrong branches while retaining all correct ones? (3) Is there any correlation between the support of a branch and the actual amount of evolution along that branch?

Under the conditions of our experiments, bootstrap and Bayesian techniques outperform a consensus of MP trees, with respect to the first question. Further, we find that the support of a correct branch as computed by each of the techniques remains largely unaffected by amount of evolution along that branch. However, with respect to the second question, the answer is not very promising. Under the conditions of our experiments, any choice of threshold for any of the methods involves a significant tradeoff between the number of wrong branches contracted and the number of correct branches retained.

2. Methods

In this study we considered three different phylogenetic estimation methods—Maximum Parsimony[9] (MP), Maximum Likelihood[7] (ML), and Bayesian Inference[10] (BI). Since MP and ML estimation methods do not produce trees that have support-labeled branches, these methods are used in conjunction with a bootstrap algorithm in order to generate support values for tree branches. Another prevalent method for generating support-labeled trees is to take the majority consensus of the top scoring trees returned by MP, which we also considered in our study.

2.1. *Phylogeny Estimation Methods*

Two of the most commonly used and most accurate criteria for phylogeny reconstruction are *maximum parsimony* (MP) and *maximum likelihood* (ML). They are both hard optimization criteria for which various accurate heuristics have been devised.

The MP criterion is based on the assumption that "evolution is parsimonious", i.e., the best evolutionary trees are the ones that minimize the number of changes along the

branches of the tree. In our study, we used the PAUP*[21] MP heuristic (starts with a random tree, and traverses the tree space using TBR moves). The ML problem seeks the tree T and its associated parameters (such as branch-lengths, rates of evolution for each site, etc.) that maximize the probability of generating the given set of sequences. In our study, we used the PAUP*[21] ML heuristic (starts with a random tree, and traverses the tree space using TBR moves). *Bayesian Inference* seeks the tree that maximizes the estimated posterior probability of the tree τ_i given the sequences $\mathbf{X}$. The MrBayes tool is a heuristic that uses the Markov chain Monte Carlo method to approximate the posterior probability.[11] We used this application for inferring trees (we used 100,000 generations with a burn-in period of 10,000 generations).

2.2. *Bootstrap*

The bootstrap technique is commonly used to add support-labelings to the output of MP and ML estimation methods. This technique subsamples the original sequence data to produce "new" input data of the same length in which some of the original sites appear duplicated and some do not appear at all. The technique constructs the new datasets in such a way that they remain statistically similar to the original input data. The bootstrap technique constructs these datasets and the associated best MP or ML tree a specified number of times. Following this, a support-labeled tree is constructed by taking the majority consensus of the set of trees created during the iterations. For our study, we used the bootstrap techniques with MP and ML, as implemented in PAUP*,[21] where the number of repetitions we considered was 100.

2.3. *Consensus Trees & Branch Contraction*

The *p-consensus tree*, τ_c, for a set of trees, T, is the tree containing only those branches that occur in at least p percent of the trees in T. Associated with each branch in the consensus tree is the percent of trees that contain that branch—this is considered the support for that branch. A *strict consensus tree* is a consensus tree for which $p = 100$. Therefore, it contains only those branches that occur in all of the trees in T. On the other end of the spectrum, the *majority consensus trees* is the consensus tree for which $p = 50$, containing only those branches that occur in at least half of the trees in T.

In a strict consensus tree, the minimum support of any branch in the tree is 100. Also the maximum support any branch can possibly have in any tree is 100. As a result, all branches in a consensus tree have the same, maximum support value.

In a majority consensus tree, the support for any branch can range between 50 and 100. After constructing such a majority consensus tree, we may want to remove all branches that have a support value below a certain threshold. This threshold-based removal procedure is called *branch contraction*. Assuming that some branches are removed by such a process, the result of branch contraction is an unresolved (non-binary) tree in which all remaining branches have a support value greater than or equal to the threshold.

2.4. *Tree Comparison*

Given two trees, the model τ_M and the estimate τ_e, the distance is reported in terms of *false positives*, the number of branches in τ_e that are not in the model τ_M, and *false negatives*, the

number of branches in τ_M that are missing from τ_e. The false negative and false positive values are divided by the number of branches in τ_e, so that both error rates fall between 0 and 1. In this study, we used the false positives (FP), false negatives (FN), and their average (also known as the Robinson-Foulds measure[16]) to quantify the error between the model and inferred trees.

3. Experimental Design

Sequence Dataset Generation. We generated five different fully resolved, 20-taxon trees using the r8s tree generation tool.[17] We then deviated each tree from ultrametricity by scaling each branch length by a random value $r = e^x$ where $-2 \leq x \leq 2$. These deviated trees were designated the model trees. For each model tree, and each sequence length of 250, 500, 1000, and 1500 nucleotides, we generated 40 DNA sequence datasets using Seq-gen with the scaling time-reversible model and a substitution rate of 0.6.[15]

True Tree Calculations. For the purpose of our study, we differentiate between the *model tree* and the *true tree*. The branch lengths of the former are the *expected* numbers of changes along the branches, whereas the branch lengths of the latter are the *actual* numbers of changes along the branches. Since we want to study the performance of support assessment techniques as a function of the actual branch lengths, we generated true trees by relabeling the branch lengths of the model trees t_M with the actual substitution rates (which are known in simulations).

Generating ML Bootstrap, MP Boostrap and BI Results. We used PAUP*[21] to generate ML and MP bootstrap results (100 repetitions), and MrBayes[11] for Bayesian inference. We ran each of the methods on each sequence dataset individually.

Generating MP Consensus Trees. Majority consensus MP trees were generated by a series of steps. First, we ran the PAUP* implementation of MP (described above) and reported all trees. We separated the trees into levels, where the top level corresponded to trees with the lowest parsimony score (the best trees), and each subsequent level contained trees of increasing parsimony score. We calculated the majority consensus trees for each sequence dataset using trees from just level 1; levels 1 and 2; levels 1,2, and 3; and levels 1,2,3, and 4.

4. Results

In order to characterize the performance of the different estimation methods, we chose to study the relationship between the substitution rate along a branch and each method's support for that branch as well as the interplay between the contraction threshold and three different measurements of tree errors (false positives, false negative, and average error). In Section 5, we compare the results of each method. The standard deviations for the results of each method were small (MP ≤ 0.084, ML ≤ 0.083, and MB ≤ 0.047) and will not be shown in figures to enhance readability.

4.1. *Selection of Optimal MP Consensus Method*

Recall from Section 3 that we generated results for MP majority consensus for four combinations of top tree levels (1; 1 and 2; 1, 2 and 3; 1, 2, 3, and 4). Therefore, for any given

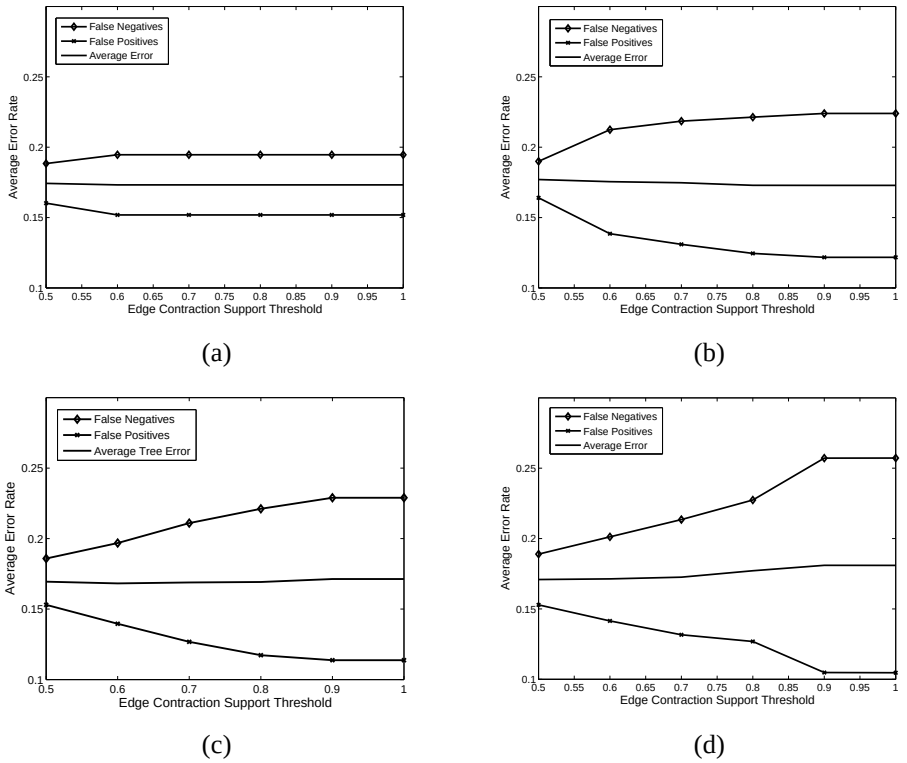

Figure 1. The average error of the estimated trees constructed using MP with Majority Consensus on the datasets with a sequence length of 1500. The x-axis is the range of possible contraction support threshold values. Errors are calculated with respect to the true tree for the dataset. The errors reported are the average of the error for each tree constructed for each dataset. The graphs show Majority Consensus using the (a) top one level, (b) top two levels, (c) top three levels, and (d) top four levels.

sequence dataset there are four MP consensus trees, corresponding to each of these level combinations. In the remainder of the analysis, we compare only the best of the four MP consensus level sets with MP bootstrap, ML bootstrap, and BI. Figure 1 shows the average performance of this method over all datsets with sequence length of 1500 for the four choices of levels. While average total error is nearly identical for all choices of trees, consensus trees built from all four levels contain the fewest false positives, yielding a tree with fewer wrong relationships than other trees. As a result, we chose the 4-level MP consensus trees to be representative of the MP consensus method in the remainder of this study. A significant observation is that regardless of the threshold value chosen, the average error rate of the majority consensus tree does not drop below 16%.

4.2. *Branch Support vs. Substitution Rate*

Within a "reasonable" range of substitution values (well below the point of saturation), it is usually the case that a larger number of substitutions along a branch is correlated to a higher

probability of inferring that branch. Therefore, branches in the true tree (whose branch lengths are well below the point of saturation in our experiments) with high substitution rates should have a stronger phylogenetic signal and hence higher probability of being inferred. We tested this hypothesis by grouping branches in the true trees by their actual substitution rate, creating five bins for branches with substitution rates in the ranges 0—0.1, 0.1—0.2, 0.2—0.3, 0.3—0.4, and 0.4—0.5. For each method, we collected the support values generated for the branches in each dataset. The resulting distributions of support values in each bin for datasets with a sequence length of 1500 are shown in Figure 2.

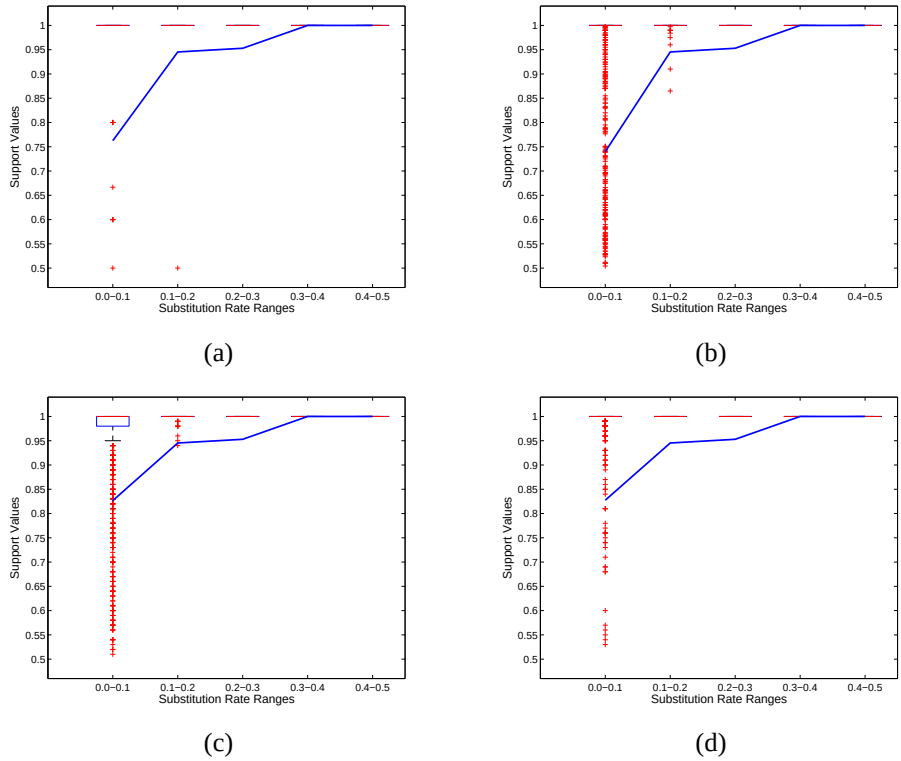

Figure 2. Whisker-box plots indicating the distribution of support values for each substitution rate range: (a) MP consensus, (b) MP bootstrap, (c) ML bootstrap, and (d) BI. The + marks indicate outliers. The trend lines indicate the percentage of correct branches (true positives) in each bin that were predicted in the estimated trees on average.

The trend lines indicate the percentage of correct branches (true positives) in the bin that were predicted by the each method. As expected, the percent of true branches predicted is higher for branches with greater substitution rates.

In Figure 2, the whisker-box plots[13] and average support values for all substitution ranges for all methods are compressed into a very small region around 1.0, indicating that the majority of support values for branches, regardless of true substitution rate or method, were close to 1. In the lower substitution rate range, these were much higher support

values than we expected to see. Surprisingly, repeating the same test on the 250 bp datasets (not shown) yields similarly high average support values – greater than 90% in the lowest substitution rate bin for all methods. Thus, MP bootstrap, ML bootstrap, and BI methods all characteristically assign high support values even to branches with low substitution rates – implying that when one of these methods detects a true branch, it obtains a strong signal, regardless of the true substitution rate along that branch. However, observe that for branches with low substitution rates, consensus and bootstrap MP trees have a false negatives rate of about 25%, and for bootstrap ML and BI trees have a false negatives rate of about 15%. In other words, while these methods are computing high support of very short branches, they are missing a sizable portion of the true branches. Further, the MP consensus trees have the least number of outliers for very short branches; yet, this comes at the expense of higher false negatives rate. The other methods have higher numbers of outliers and lower false negative rates.

4.3. *Effects of Branch Contraction on Accuracy*

The benefit of having support-weighted branches is that branches with low support (defined as appropriate for the intended use of the tree) can be removed through branch contraction. In order to characterize how branch contraction can be used to derive more accurate phylogenetic trees, we calculated the error of each estimated support-labeled tree for various choices of a branch contraction threshold. These results are shown in Figure 3. The figure shows the error measured in false positives, false negatives, and average error (as defined in Section 2) for all four methods. There are several trends evident in the graphs:

False positives monotonically fall with higher branch contraction thresholds. This trend can be seen in all four plots shown in Figure 3 and is expected since we assume that the noise in the data giving rise to the prediction of incorrect branches is minimal, leading to those wrong branches having small support values. This is precisely what is observed.

Low-supported branches are evenly split between true and false positives. Despite the fact that the number of false positives falls with higher branch contraction thresholds, the number of false negatives rises. On all four plots, the slope of the false positives line is mirrored by the slope of the false negatives line. This indicates that approximately as many true branches receive low support values as do false branches. An ideal method would have a falling false positive score and a constant false negative score for increasing contraction thresholds.

Overall average error modestly increases with higher branch contraction thresholds. Due to the fact that as the branch contraction threshold increases, the false positives decrease and the false negatives increase at similar rates, we expect that the overall error will not change significantly. In fact, for all methods, as the contraction threshold is increased, the average error increases slightly, seen most prominently in the MP bootstrap (Figure 3(b)) and ML boostrap (Figure 3(c)) methods. This should not be interpreted as implying that the trees are of equal correctness. On the contrary, as will be discussed in Section 5, the overall average error is not the best error metric to use when evaluating the correctness of a tree.

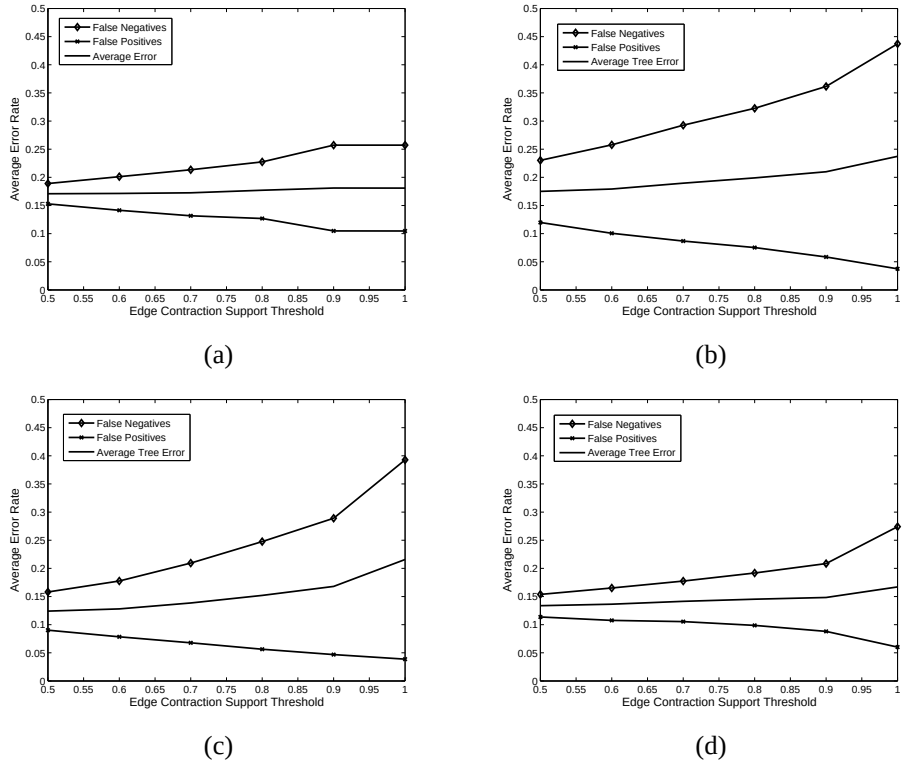

Figure 3. The average error of estimated trees, branch contracted according to the threshold shown along the x-axis, as compared to the true tree for the dataset: (a) MP consensus, (b) MP bootstrap, (c) ML bootstrap, and (d) BI. All figures shown were constructed using only the results from the 1500 sequence length datasets.

The MP consensus method has few moderately supported branches. Unlike the MP bootstrap, ML boostrap, and BI methods (Figures 3(b—d)), the numbers of false positives and negatives for the MP consensus method hardly change for different values of the contraction threshold. This is an indicator that branches in MP concensus trees characteristically have extreme support values - either close to 0.5 or 1 - resulting from the population of trees is generally too small in size to provide sufficient diversity to generate good support values. In spite of this limitation, the MP consensus method still generates trees with comparable overall average error levels, albeit with undesirably high false positive rates for high contraction thresholds.

5. Discussion

The overarching goal of this project was to find out how support values generated by various phylogenetic estimation methods can be used to estimate more accurate trees. Based on the results presented in Section 4, specifically those discussed in Section 4.3, we have several observations (true under the conditions of our experiments) to offer:

The MP consensus method does not produce informative support-labeled trees. Though

the method does produce good trees in terms of all three forms of error measured in this project, Figure 3(a) reveals that support-values cannot be used to significantly improve the majority consensus tree.

MP bootstrap, ML bootstrap, and BI perform very similarly. BI produces the most resolved trees of the three methods (evident from its significantly lower false negative rates), whereas MP and ML both have slightly lower false positive rates (which are less significant than the false negative difference in BI).

Strict consensus gives the most correct tree. We make this observation from the perspective of minimizing the false positives. As discussed earlier in Section 2, false negatives lead to conservative trees, missing some resolution in the relationships between taxa whereas false positives are relationships that do not actually exist. While Figure 3 shows that strict consensus trees will contain more errors than majority consensus trees, the strict consensus trees will be conservative estimates as opposed to majority consensus which contain wrong relationships.

It is impossible to construct a fully resolved (binary) tree with 100% certainty. As the figures show, attempting to maintain a more resolved tree requires the admission of more false positives into the tree. In order to eliminate these errors, the tree must become less resolved. Because of this trade-off, phylogenetic analysis methods must be designed to operate on non-binary trees. The alternative is to accept greater accumulated error in the results.

6. Conclusions

In this project, we studied four different phylogenetic estimation methods for constructing support-labeled trees. The contribution of this paper bears a significant impact on the understanding of the relative merits of the different algorithms we studied and of the trade-off involved in choosing a branch contraction threshold. In addition, our results support the observation that strict consensus trees will always yield more correct trees, if the goal is to minimize the number of wrong branches in the estimated tree. Further, our results show that even with sophisticated methods such as Bayesian inference, obtaining a fully resolved accurate tree is very hard. Therefore, phylogenetic analysis tools that assume the trees are always binary (fully resolved) may have a serious shortcoming in their applicability.

This study has also identified the trend of methods ascribing low support values to equal numbers of true and false branches in the estimated tree. What remains unclear is why true branches receive low support values and whether there are ways to improve this true branch confidence. Such improvements would directly impact the accuracy of estimated trees.

References

1. M. Alfaro, S. Zoller, and F. Lutzoni. Bayes or Bootstrap? A Simulation Study Comparing the Performance of Bayesian Markov Chain Monte Carlo Sampling and Bootstrapping in Assessing Phylogenetic Confidence. *Mol. Biol. Evol.*, 20(2):255–266, 2003.
2. V. Berry and O. Gascuel. On the Interpretation of Bootstrap Trees: Appropriate Threshold of Clade Selection and Induced Gain. *Mol. Biol. Evol.*, 13(7):999-1011, 1996.
3. V. Berry, O. Gascuel, and G. Caraux. Choosing the tree which actually best explains the data: another look at the bootstrap in phylogenetic reconstruction. *Computational Statistics & Data Analysis*, 38:273–283, 2000.

4. V. Berry, D. Bryant, T. Jiang, P. Kearney, M. Li, T. Wareham, and H. Zhang. A Practical Algorithm for Recovering the Best Supported Edges of an Evolutionary Tree. In *Proc. 11th Annual ACM-SIAM Symposium on Discrete Algorithms*, 287-296, 2000.
5. M. P. Cummings, S. A. Handley, D. S Myers, D. L. Reed, A. Rokas, and K. Winka. Comparing Bootstrap and Posterior Probability Values in the Four-Taxon Case. *Syst. Biol.*, 52(4):477–487, 2003.
6. P. Erixon, B. Svennblad, T. Britton, and B. Oxelman. Reliability of Bayesian Posterior Probabilities and Bootstrap Frequencies in Phylogenetics. *Syst. Biol.*, 52(5):665–673, 2003.
7. J. Felsenstein. Evolutionary trees from DNA sequences: a maximum likelihood approach. *Journal of Molecular Biology*, 17:368–376, 1981.
8. J. Felsenstein. Confidence limits on phyogenies: an approach using the bootstrap. *Evolution*, 39:783-791, 1985.
9. J. Felsenstein. *Inferring Phylogenies*. Sinauer Associates, Inc., Sunderland, MA, 2003.
10. J. Huelsenbeck, B. Rannala, J. Masly. Accomodating phylogenetic uncertainty in evolutionary studies. *Science*, 288:2349–2350, 2000.
11. J. Huelsenbeck and F. Ronquist. MrBayes: Bayesian inference of phylogenetic trees. *Bioinformacs*, 17(8):754–755, 2001.
12. A. Leache and T. Reeder. Molecular Systematics of the Eastern Fence Lizard (Sceloporus undulatus): A comparison of Parsimony, Likelihood, and Bayesian Approaches. *Syst. Biol.*, 46:523–536, 2002.
13. R. McGill, J. W. Tukey, and W. A. Larsen. Variations of Boxplots. *The American Statistician*, 32:12-16, 1978.
14. M. Nei, S. Kumar, and K. Takahashi. The optimization principle in phylogenetic analysis tends to give incorrect topologies when the number of nucleotides or amino acids is small. *Proc. Natl. Acad. Sci. USA*, 95: 12390–12397, 1998.
15. A. Rambaut and N. C. Grassly. Seq-Gen: an application for the Monte Carlo simulation of DNA sequence evolution along phylogenetic trees. *Comput. Appl. Biosci.*, 13:235–238, 1997.
16. D.R. Robinson and L.R. Foulds. Comparison of phylogenetic trees. *Math. Biosciences*, 53:131–147, 1981.
17. M. Sanderson. r8s software package, Available from http://loco.ucdavis.edu/r8s/r8s.html, 2001.
18. M. Simmons, K. Pickett, and M. Miya. How Meaningful AER Bayesian Support Values? *Mol. Biol. Evol.*, 21(1): 188–199, 2004.
19. P. Soltis and D. Soltis. Applying the Bootstrap in Phylogeny Reconstruction. *Statistical Science*, 18(3):256–267, 2003.
20. Y. Suzuki, G. Glazko, and M. Nei. Overcredibility of molecular phylogenies obtained by Bayesian phylogenetics. *PNAS*, 99(25): 16138–16143, 2002.
21. D. Swofford. PAUP*: Phylogenetic analysis using parsimony (*and other methods). Version 4.0b10. Sinauer Associates, Sunderland, Mass, 2002.
22. K. Takahashi and M. Nei. Efficiencies of Fast Algorithms of Phylogenetic Inference Under the Criteria of Maximum Parsimony, Minimum Evolution, and Maximum Likelihood When a Large Number of Sequences Are Used. *Mol. Biol. Evol.*, 17(8):1251–1258, 2000.
23. D. Taylor and W. Piel. An Assessment of Accuracy, Error, and Conflict with Support Values from Genome-Scale Phylogenetic Data. *Mol. Biol. Evol.*, 21(8):1534–1537, 2004.