bioengineering

MDPI

Review

Common Methods for Phylogenetic Tree Construction and Their Implementation in R

Yue Zou [1,†], Zixuan Zhang [1,†], Yujie Zeng [1], Hanyue Hu [1], Youjin Hao [1], Sheng Huang [2,*] and Bo Li [1,*]

[1] College of Life Sciences, Chongqing Normal University, Chongqing 401331, China; 2021051301110@stu.cqnu.edu.cn (Y.Z.); 2020051305046@stu.cqnu.edu.cn (Z.Z.); zengyj2000118@163.com (Y.Z.); 2023210513008@stu.cqnu.edu.cn (H.H.); haoyoujin@hotmail.com (Y.H.)
[2] Animal Nutrition Institute, Chongqing Academy of Animal Science, Chongqing 402460, China
[*] Correspondence: stevenhouse@alu.cqu.edu.cn (S.H.); libcell@cqnu.edu.cn (B.L.)
[†] These authors contributed equally to this work.

Abstract: A phylogenetic tree can reflect the evolutionary relationships between species or gene families, and they play a critical role in modern biological research. In this review, we summarize common methods for constructing phylogenetic trees, including distance methods, maximum parsimony, maximum likelihood, Bayesian inference, and tree-integration methods (supermatrix and supertree). Here we discuss the advantages, shortcomings, and applications of each method and offer relevant codes to construct phylogenetic trees from molecular data using packages and algorithms in R. This review aims to provide comprehensive guidance and reference for researchers seeking to construct phylogenetic trees while also promoting further development and innovation in this field. By offering a clear and concise overview of the different methods available, we hope to enable researchers to select the most appropriate approach for their specific research questions and datasets.

Keywords: phylogenetic tree; neighbor-joining method; maximum parsimony method; maximum likelihood method; Bayesian method; tree integration; R language

check for updates

Citation: Zou, Y.; Zhang, Z.; Zeng, Y.; Hu, H.; Hao, Y.; Huang, S.; Li, B. Common Methods for Phylogenetic Tree Construction and Their Implementation in R. *Bioengineering* **2024**, *11*, 480. https://doi.org/10.3390/bioengineering11050480

Academic Editor: Kwang Woo Ahn

Received: 4 April 2024
Revised: 4 May 2024
Accepted: 7 May 2024
Published: 11 May 2024

1. Introduction

A phylogenetic tree, also known as a cladogram, tree of life, or evolutionary tree, is a graphical representation resembling a tree that illustrates the evolutionary and phylogenetic relationships between biological taxa based on their physical or genetic characteristics [1–3]. Comprising nodes and branches, a phylogenetic tree uses nodes to stand for taxonomic units and branches to depict estimated time relationships between these units [4,5]. As shown in Figure 1, there exist two types of nodes in a phylogenetic tree: internal nodes and external nodes (leaf nodes). Internal nodes are hypothetical taxonomic units (HTUs), with the topmost internal node called the root node, symbolizing the most recent common ancestor of all leaf nodes, marking the starting point of evolution. External nodes represent operational taxonomic units (OTUs), typically indicating species but also capable of representing extinct lineages or fossil endpoints [6,7]. The evolutionary clade within the phylogenetic tree encompasses a node and all lineages stemming from it. Depending on the different topological structures, phylogenetic trees can be categorized into rooted trees and unrooted trees: rooted trees have a root node from which the rest of the tree diverges, indicating an evolutionary direction. In contrast, unrooted trees lack a root node and only illustrate relationships between nodes without suggesting any evolutionary direction [8].

The phylogenetic tree visually presents the evolutionary history and phylogenetic relationships between different taxonomic units, facilitating people's understanding of the causes of species' morphological diversity and evolutionary patterns [9]. On the one hand, a phylogenetic tree can drive the development of phylogenetic systematics [10]. On the other hand, it can help reveal patterns such as genetic structure, gene flow, and genetic drift among populations, providing important clues for population genetics research [11,12].

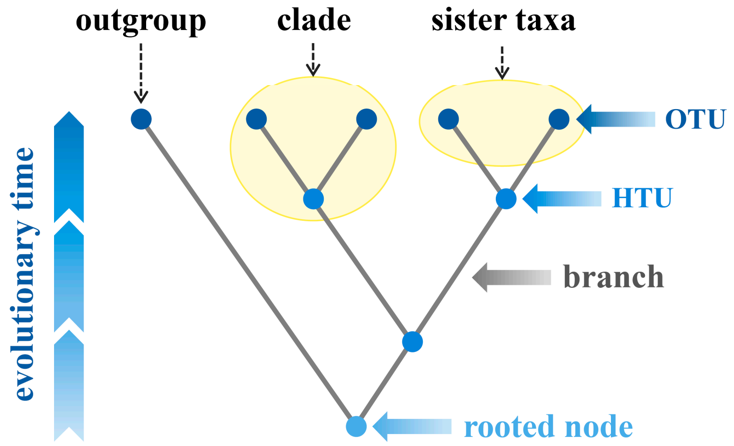


Figure 1. The general structure of a phylogenetic tree. The abbreviations in the figure are as follows: OTU, operational taxonomic unit; HTU, hypothetical taxonomic unit. All figures in this review were drawn by Yue Zou, using Microsoft PowerPoint 2010 and Adobe Illustrator 26.2.1.

2. The Popular Methods for Inferring Phylogenetic Trees

Before the advent of DNA sequencing technologies, biologists typically relied on traditional taxonomic features such as biological morphology and traits to infer phylogenetic trees (regarded as species trees). However, with the development of sequencing technologies, a large amount of accumulated gene sequences became the basic data for inferring species trees [13,14]. Since genes and species often coevolve, they often exhibit similar evolutionary patterns, allowing gene trees to be used for inferring species trees [15,16]. Figure 2 illustrates the general process of constructing a phylogenetic tree starting from gene sequences, including steps such as sequence collection, sequence alignment, model selection, tree inference, and tree evaluation. Typically, researchers first collect homologous DNA (or protein) sequences through experiments or public databases (such as GenBank, EMBL, DDBJ) and then perform sequence alignment. Accurate alignment results form the basis for inferring evolutionary relationships, and multiple methods are commonly used in practice to generate consistent results [17]. It should be noted that the aligned sequences need to be precisely trimmed before inferring the tree structure to remove unreliable regions that may affect subsequent analysis [18]. Insufficient trimming may introduce noise, while excessive trimming may remove genuine signals that help with phylogenetic analysis [19,20]. Once the sequence alignment is completed, researchers then select appropriate algorithms for phylogenetic tree inference [21].

There are two main categories of methods used for phylogenetic tree inference [22]: (1) Distance-based methods (such as the NJ method and the UPGMA method) [23]. These methods first convert the feature matrix into a distance matrix to represent the evolutionary distances between pairs of species, and then combine clustering algorithms to analyze the species under study and infer the phylogenetic tree [24]. (2) Character-based methods (such as the parsimony method and the likelihood method) [25,26]. These methods typically generate a large number of hypothetical trees based on an algorithm (such as the MP method, ML method, and BI method) and then induce an optimal tree according to certain criteria [27]. Among them, the parsimony method has no explicit model assumptions, while the likelihood method has a specific fixed sequence evolution model and likelihood function [28]. Distance methods always produce a single evolutionary tree, while parsimony and likelihood methods involve numerous hypothetical trees before producing the optimal tree. The characteristics of these common tree-building methods are shown in Table 1.

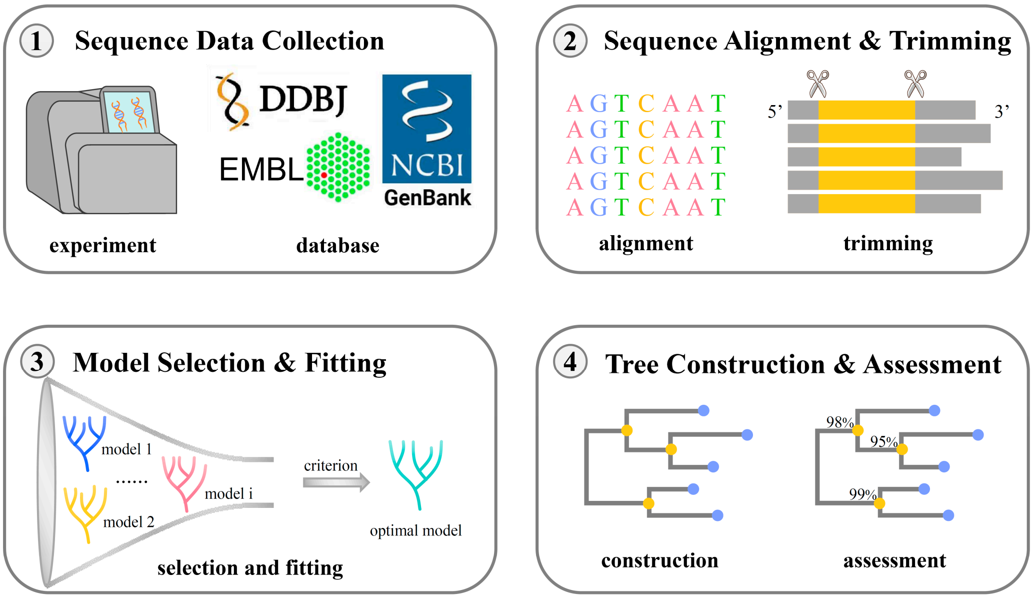


Figure 2. The construction workflow of a phylogenetic tree. A classic process used to build the evolutionary tree usually contains the following steps: (**1**) sequence data collection, (**2**) sequence alignment and trimming, (**3**) model selection and fitting, as well as (**4**) tree construction and evaluation. The abbreviations in the figure are as follows: DDBJ, DNA Data Bank of Japan; EMBL, European Molecular Biology Laboratory; NCBI, National Center for Biotechnology Information.

Table 1. The common algorithms used in phylogenetic tree construction.

Algorithm	Principle	Hypothesis	Criteria for Selecting the Final Tree	Scope of Application
NJ *	Minimal evolution: Minimizing the total branch length of the phylogenetic tree.	BME branch length estimation model: Ensuring general statistical consistency of minimum length phylogeny and non-negativity of its branch lengths [21].	In the end, only one tree was constructed.	Short sequences with small evolutionary distance and few informative sites.
MP	Maximum-parsimony criterion: Minimize the number of evolutionary steps required to explain the data set.	No model required.	The phylogenetic tree with the smallest number of base (or amino acid) substitutions during evolution.	Sequences with high sequence similarity, sequences for which it is difficult to design appropriate characteristic evolution models.
ML	Maximize likelihood value.	The sites in the alignment are independent; each branch is allowed to evolve at different rates.	Phylogenetic tree with maximum likelihood value.	Distantly related and small number of sequences.
BI	Bayes theorem.	Continuous-time Markov substitution model: Substitution probability is only related to the current nucleotide and has nothing to do with past nucleotides.	The most sampled phylogenetic tree in MCMC.	A small number of sequences.

* NJ: a representative method and one of the most popular distance-based methods. The abbreviations in the table are as follows: NJ, neighbor-joining; MP, maximum parsimony; ML, maximum likelihood; BI, Bayesian inference; MCMC, Markov chain Monte Carlo.

2.1. Distance-Based Method

Distance-based methods are the simplest approach for constructing phylogenetic trees. They transform the molecular feature matrix of different species into a distance matrix and then use clustering algorithms to classify these species and infer the evolutionary relationships [29]. Representative methods in this category include neighbor-joining (NJ) and unweighted pair group method with arithmetic mean (UPGMA) [30]. The NJ method, created by Naruya Saitou and Masatoshi Nei in 1987, is an agglomerative clustering algorithm [31]. The tree-building process is illustrated in Figure 3. Firstly, an initial distance matrix is constructed based on similarity measures between sequences. In practice, users can choose appropriate distance metrics (such as the Hamming distance, Jaccard distance, Euclidean distance, and Manhattan distance) according to the characteristics of the sequence data and the research question. Then, an initial tree for an unrooted star-like network is created based on the initial matrix. Subsequently, the distance matrix is updated by merging the two nodes with the smallest distance, and a new node connecting these two clusters is created in the tree topology. This new node is connected to the central node, updating the tree topology. This step is repeated until only one cluster remains, resulting in the NJ tree.

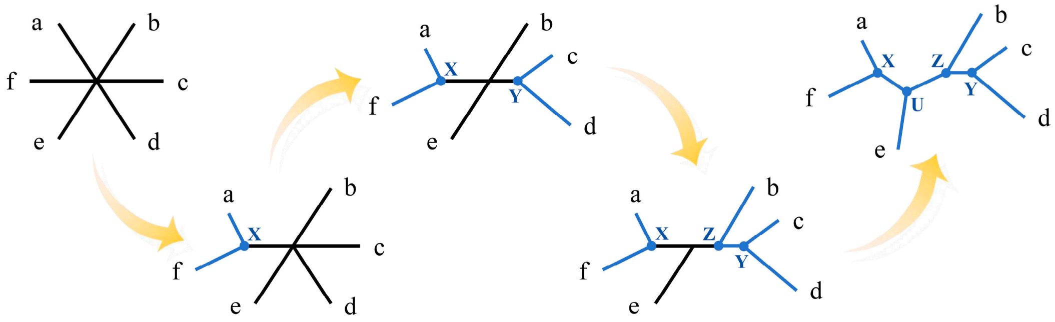
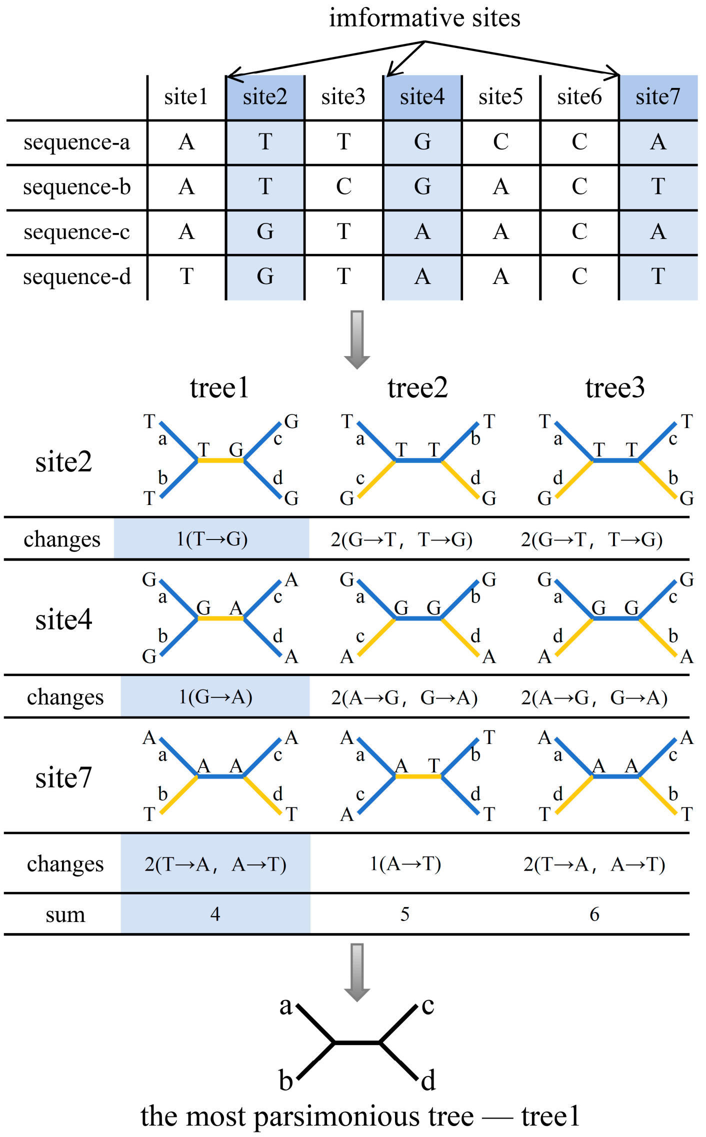


Figure 3. Fundamentals of the neighbor-joining methods for constructing phylogenetic trees. The abbreviations in the figure are as follows: a, b, c, d, e, and f represent different operational taxonomic units, and X, Y, Z, and U represent different hypothetical taxonomic units.

The NJ method has high accuracy and fewer assumptions when reconstructing phylogenetic trees. It also has a faster computation speed. It uses a stepwise construction approach to build the evolutionary tree instead of searching for the optimal tree [32,33]. As the number of sequences increases, the number of potential topologies to be examined grows exponentially, making the probability of finding the best tree rapidly decrease. At this point, the advantages of the NJ method over the parsimony method and likelihood method become more evident, leading to its wide usage in analyzing large datasets [34]. Additionally, the neighbor-joining method allows for different branch lengths between sequences and permits multiple substitutions. However, converting sequence differences into a distance matrix may result in a reduction of sequence information when the sequence divergence is substantial [35].

2.2. Maximum Parsimony (MP) Method

Maximum parsimony (MP) is a phylogenetic tree reconstruction algorithm based on the principle of Occam’s razor, aiming to infer the evolutionary tree by minimizing the number of evolutionary steps required to explain the dataset [36]. This method was proposed by James S. Farris and Walter M. Fitch in 1970–1971 [37,38]. MP primarily considers informative sites and requires the identification of informative sites in the sequences before tree construction [39]. Figure 4 illustrates the basic process of constructing an evolutionary tree using MP. Taking DNA sequences as an example, a site is considered informative if it has at least two different nucleotides and each nucleotide appears in at least two of the studied sequences. By using informative sites, all possible tree topologies (constructing the tree space) are searched, and the minimum number of nucleotide substitutions for each topology is counted to obtain the most parsimonious tree. In simple terms, it involves finding the tree that minimizes the total number of substitutions across all informative sites. Increasing the number of taxa during tree construction leads to a rapid increase in the number of possible tree topologies. Therefore, when there are fewer taxa, exhaustive search algorithms are often used, while branch-and-bound and heuristic search algorithms are used to improve computational efficiency when there are more taxa [35]. Popular heuristic algorithms include Subtree Pruning and Regrafting (SPR) and Nearest Neighbor Interchange (NNI) [40]. MP may result in multiple equally parsimonious trees, so it is common practice to construct a consensus tree to represent the final result. This is achieved by treating consistent branch points in all trees as binary branches, converting partially consistent branch points into internal nodes connecting multiple branches, or selecting the most frequently occurring branch points among all MP trees [35].



	site1	site2	site3	site4	site5	site6	site7
sequence-a	A	T	T	G	C	C	A
sequence-b	A	T	C	G	A	C	T
sequence-c	A	G	T	A	A	C	A
sequence-d	T	G	T	A	A	C	T

	tree1	tree2	tree3
site2 changes	1(T→G)	2(G→T， T→G)	2(G→T， T→G)
site4 changes	1(G→A)	2(A→G， G→A)	2(A→G， G→A)
site7 changes	2(T→A， A→T)	1(A→T)	2(T→A， A→T)
sum	4	5	6

Figure 4. Fundamentals of the maximum parsimony method for constructing phylogenetic trees. The abbreviations in the figure are as follows: a, b, c, and d represent sequences a, b, c, and d, respectively.

Maximum parsimony is known for its straightforward mathematical approach and absence of a specific model. It is well suited for data types where designing appropriate evolutionary models is challenging, such as rare features based on genomic rearrangements or unique morphological traits. However, when applied to large datasets, it frequently generates numerous potential rooted trees, rendering comprehensive comparisons unfeasible [28].

2.3. Maximum Likelihood (ML) Method

Maximum likelihood (ML) was first proposed by Felsenstein in the early 1980s [41]. The main process of constructing an evolutionary tree using this method is shown in Figure 5.

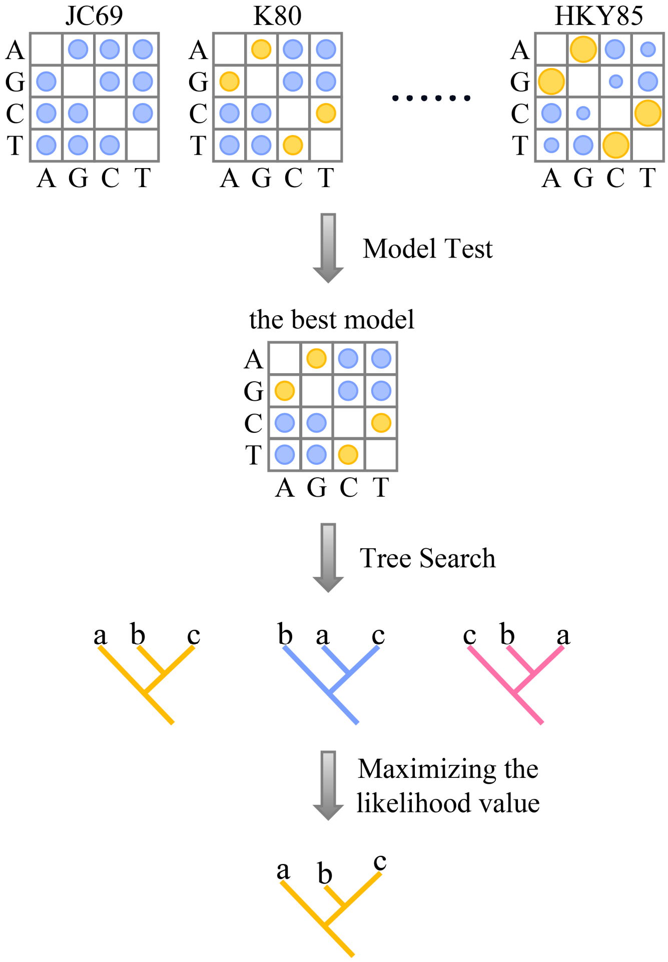


Figure 5. Fundamentals of maximum likelihood method for constructing phylogenetic trees. The abbreviations in the figure are as follows: JC69, the Jukes and Cantor 1969 model; K80, the Kimura 1980 model; HKY85, the Hasegawa, Kishino, and Yano 1985 model; a, b, and c represent different species.

First, a suitable evolutionary model is selected based on the characteristics of the sequence data being studied. JC69 [42,43], K80 [44], TN93 [45], HKY85 [46], and GTR [47] are commonly used evolutionary models for analyzing DNA sequences. The JC69 model assumes that all nucleotide substitutions occur with equal probability [42,43]. In contrast, the TN93 model assumes that transitions and transversions occur at different rates, and base frequencies are estimated from the data [48]. The GTR model assumes that all nucleotides occur at different frequencies and convert at different rates. Next, a tree space search is conducted, and optimal substitution parameters and branch lengths for each topology are optimized based on standard numerical optimization principles to maximize the likelihood value for each topology [49,50]. Finally, the topology with the highest ML value is selected as the optimal evolutionary tree. In principle, this step must be repeated for all possible trees to find the maximum likelihood value, but the number of hypothetical trees with n taxa increases rapidly with n. This means that exhaustive searches are only suitable for phylogenetic inference based on a small number of taxa, and for inference based on more taxa, tree space searches are usually heuristic [49].

Because likelihood methods have clear model assumptions, the probability of systematic errors (such as long-branch attraction artifacts) is lower than that of parsimony methods. However, the complex model settings greatly increase the computational burden. Maximum likelihood is a statistical method based on evolutionary models. It has advantages such as statistical consistency, robustness, and the ability to compare different trees and make full use of original data within a statistical framework.

2.4. Bayesian Inference (BI) Method

Bayesian inference (BI) for phylogenetic inference was proposed by Bruce Rannala and Ziheng Yang in the 1990s [51,52]. Its appearance changed the way people analyze genomic sequences [53]. Unlike ML methods, Bayesian methods use statistical distributions to quantify uncertainty in parameters [28]. The main process of tree construction is shown in Figure 6. First, a suitable evolutionary model is selected for the sequence being studied, and parameter prior information (such as tree topology and branch length) is reasonably set based on professional knowledge and experience [54]. Most phylogenetic models use continuous-time Markov processes (CTMPs) to model nucleotide substitution, which have an important property called the Markov property: the future state (remaining time before the next substitution and the character state produced by the next substitution) depends only on the current state and is independent of the past states [49].

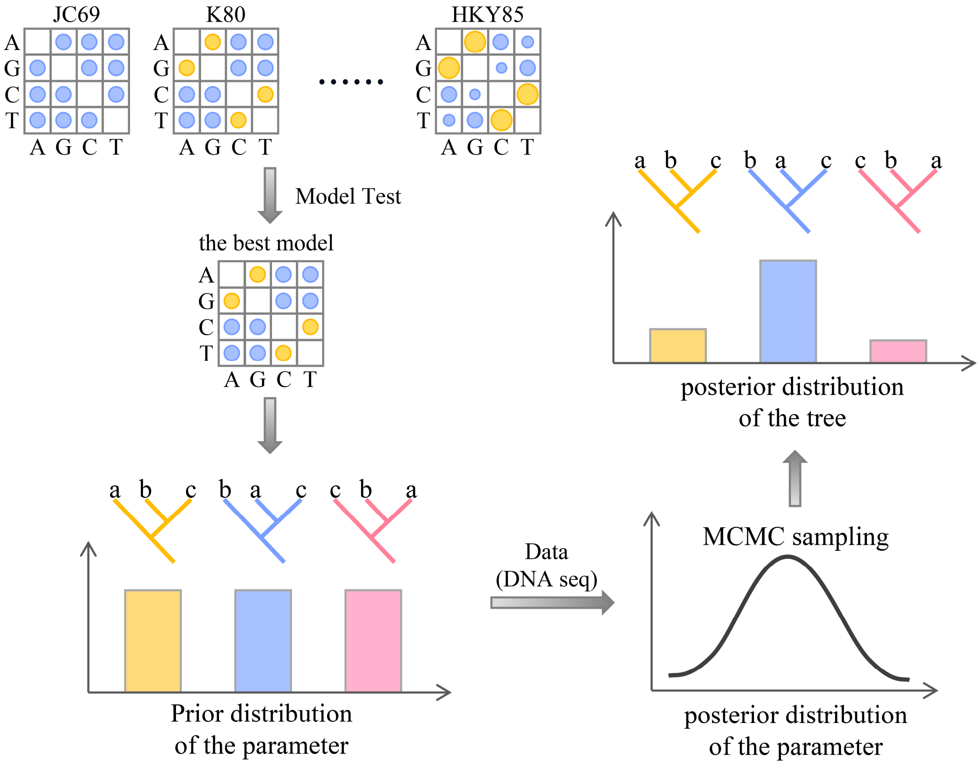


Figure 6. The principles of the Bayesian inference method for constructing phylogenetic trees. The abbreviations in the figure are as follows: JC69, the Jukes and Cantor 1969 model; K80, the Kimura 1980 model; HKY85, the Hasegawa, Kishino, and Yano 1985 model; MCMC, Markov chain Monte Carlo; a, b, and c represent different species.

According to Bayes' theorem, combining the prior information of parameters with the likelihood of sequence data can obtain posterior information of parameters, i.e., the posterior probability distribution of parameters. Then, MCMC sampling is conducted: random samples of parameters are obtained from the posterior probability distribution, and a phylogenetic tree is constructed based on each sample. This set of samples forms a

Markov chain, which converges to a stationary distribution that is equal to the posterior distribution. The most commonly used MCMC algorithms include the Metropolis–Hastings algorithm [55], Metropolis-coupled MCMC [56], and Larget and Simon's LOCAL algorithm [57]. The posterior probability distribution of trees can be approximated by the proportion of times each tree is sampled during MCMC sampling. Similarly, the posterior probability of a branch can be estimated by the proportion of sample trees that include that branch [58]. Finally, the topology with the highest posterior probability is selected as the optimal tree.

The superiority of Bayesian inference lies in its ability to handle large datasets at a higher computational speed than maximum likelihood methods and to measure the confidence of trees through posterior probabilities.

3. Advanced Computational Integrative Methods for Inferring Phylogenetic Tree

When constructing phylogenetic trees, some researchers construct trees based on individual gene (or protein) sequences, while others combine multiple gene (or protein) sequences to build a phylogenetic tree together [59,60]. For a specific group of species, phylogenetic trees constructed from individual genes often show inconsistency with each other [61,62]. As the number of taxa increases, single-gene phylogenetic trees typically have low statistical support [63]. Studies have reported that when using the same parameters and the same program, approximately 9% to 18% of single-gene phylogenetic trees cannot replicate the same topology [64].

Different genes have different evolutionary rates and evolutionary times and contain varying amounts of informative sites with different resolutions. Combining multiple gene fragments (loci) for phylogenetic analysis can provide more accurate information and higher tree resolution compared to analyzing a single gene [65,66], so the combination of multiple gene sequences has become the mainstream approach in phylogenetic studies [67]. Currently, there are two main methods for constructing multi-gene phylogenetic trees: concatenation phylogeny and coalescence phylogeny [28,68]. The prerequisite for implementing these methods is sequence alignment, and the main processes are shown in Figures 7 and 8, respectively.

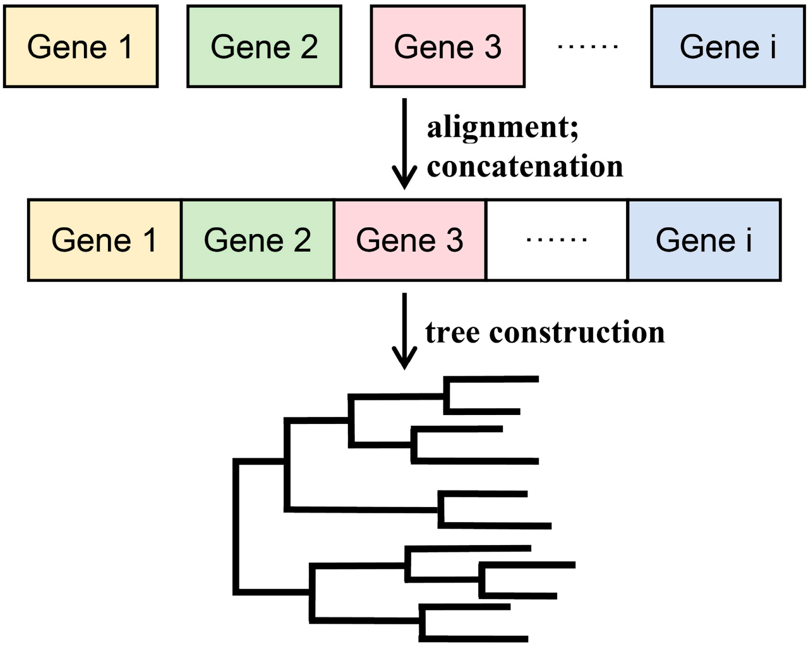


Figure 7. Strategies and principles of the concatenation phylogeny method for constructing phylogenetic trees.

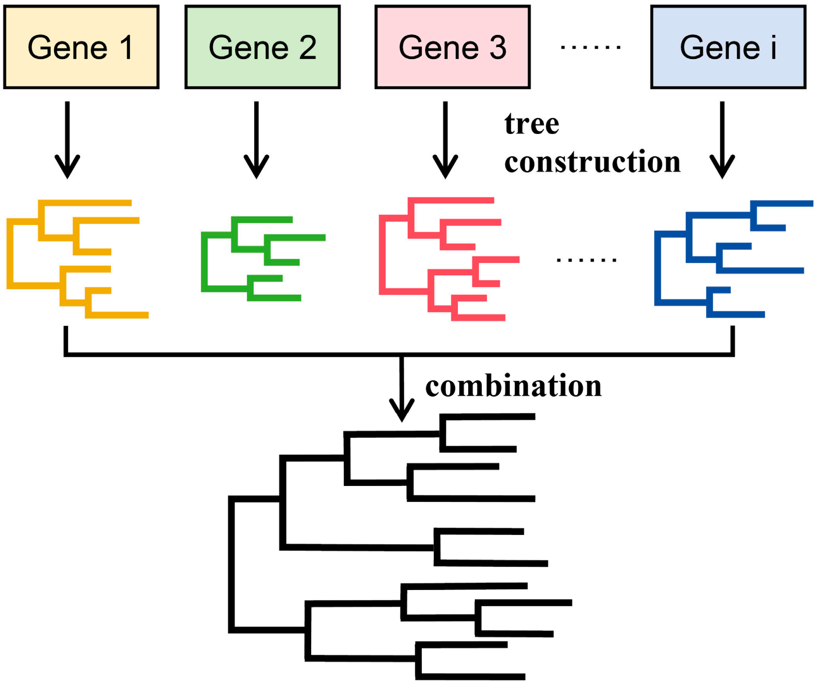


Figure 8. Strategies and principles of the coalescence phylogeny method for constructing phylogenetic trees.

3.1. Concatenation Phylogeny Method

As shown in Figure 7, concatenation phylogeny, also known as the supermatrix method or total evidence, is a method of constructing a phylogenetic tree by concatenating different gene sequences that have been aligned into a supergene matrix [69]. When following the principle of using all available data (total evidence), the most popular strategy is to use the standard molecular sequence-based methods to analyze single-gene concatenations into a "super gene" [70]. Combining multiple gene fragments for phylogenetic analysis can provide more accurate information than analyzing a single gene and can reveal hidden phylogenetic information in the data [71]. The supermatrix method fully utilizes all information from each dataset and is stable against missing data [72], but implementing this method requires all genes to have the same set of taxa. In addition, the supermatrix method usually assumes that all genes have undergone the same evolutionary process, while there may be lineage sorting during the evolution of species, which can lead to conflicts between gene trees and species trees [71].

3.2. Coalescence Phylogeny Method

As shown in Figure 8, the coalescence phylogeny method, also known as the supertree method or separate analysis, first independently analyzes each aligned gene to provide estimates for single gene trees. These individual subset trees are then integrated into a single phylogenetic tree to represent the final phylogenetic analysis. The integrated evolutionary tree will include all taxa from the source data set. Commonly used methods for integrating trees include the matrix representation with parsimony (MRP) [73,74], strict consensus [75,76], semi-strict consensus [77,78], and average consensus procedure [79,80], among which MRP is the most popular.

Unlike the supermatrix method, the supertree method only requires a partial overlap of taxa between different data sets. In cases of incomplete sampling, the results obtained from the supertree method are usually superior to those from the supermatrix method [81]. Studies across multiple data sets have shown that multi-species supertree models generally outperform concatenation models in phylogenetic inference [82]. However, because the supertree method directly operates on the phylogenetic tree and utilizes tree information summarized from various data sets, it often overlooks a significant amount of phylogenetic information [71].

4. Construction and Evaluation of Phylogenetic Trees in R Language Environment

The construction of phylogenetic trees can be achieved using various methods such as local software, online tools, and programmable code. Currently popular tree-building software includes PHYLIP [83], PAUP* [84], PhyML [85], MrBayes [86], MEGA [87], and Phylosuite [88]. These software packages typically include multiple algorithms, models, and related analysis functions (such as model comparison and bootstrap analysis), making it easier for users to perform different types of phylogenetic tree reconstruction and evolutionary analysis. However, due to the preset nature of software functionalities and options, it is often challenging to meet users' flexible analysis needs. Processing large datasets with these software tools can be cumbersome and slow, leading to various inconveniences.

In contrast, scientific programming languages like R and Python provide rich scientific computing and data analysis libraries. R is an open-source software used for statistical analysis and graphic plotting [89,90], making it particularly suitable for phylogenetic tree construction, visualization, and in-depth analysis [91]. With an extensive and vibrant user community, R fosters a collaborative environment for exchanging support, sharing experiences, and solving problems among its members. R offers a wide range of packages tailored for phylogenetic tree construction and analysis, including popular packages such as ape [92], phangorn [93], and dendextend [94], giving users greater convenience and flexibility. There is also a wealth of R packages dedicated to algorithm selection, method exploration, robust data processing, and visualization capabilities. Notable examples include Treeio [95] and tidytree [96], which facilitate the manipulation of evolutionary trees and associated data within R. In addition, ggtree [97] and ggplot2 [98] serve to enhance the visual aesthetics of phylogenetic trees while maintaining their interpretive clarity. Using the R environment for phylogenetic tree construction allows users to customize parameters to suit specific research needs.

In addition to R, Python stands out as a powerful tool for performing phylogenetic analysis, with an extensive set of libraries and tools such as Biopython [99] and DendroPy [100] that provide a wide range of functions and algorithms. In addition, Python provides access to machine learning and deep learning libraries such as scikit-learn [101] and PyTorch [102], which facilitate effective model building and prediction in phylogenetic analysis. By programming automated reconstruction of phylogenetic trees and batch processing of large datasets, researchers can significantly improve the efficiency of their analyses. It should be noted that this approach can involve more coding and debugging and requires a certain level of programming skills, which can be a learning curve for non-experts.

Figure 9 shows the whole procedure for building the phylogenetic trees in the R environment. In this review, we have selected sixteen model species, including *Homo sapiens* (human), *Mus musculus* (mouse), *Rattus norvegicus* (rat), *Drosophila melanogaster* (fruit fly), *Arabidopsis thaliana* (thale cress), *Saccharomyces cerevisiae* (brewer's yeast), *Macaca mulatta* (rhesus monkey), *Caenorhabditis elegans* (roundworm), *Sus scrofa* (pig), *Bos taurus* (cow), *Gallus gallus* (chicken), *Zea mays* (corn), *Oryza sativa* (rice), *Escherichia coli* (*E. coli*), *Glycine max* (soybean), and *Xenopus laevis* (African clawed frog). We will download their orthologous gene K00927 sequences and use them as examples to construct phylogenetic trees using the NJ method, MP method, ML method, and BI method.

Firstly, the DNA sequence files of the orthologous gene K00927 for the mentioned 16 model organisms were downloaded from the gene database under NCBI. After sequence alignment, we use the *fasta2DNAbin()* function in the R package adegenet [103] to read alignments with the FASTA format and convert them into DNAbin objects.

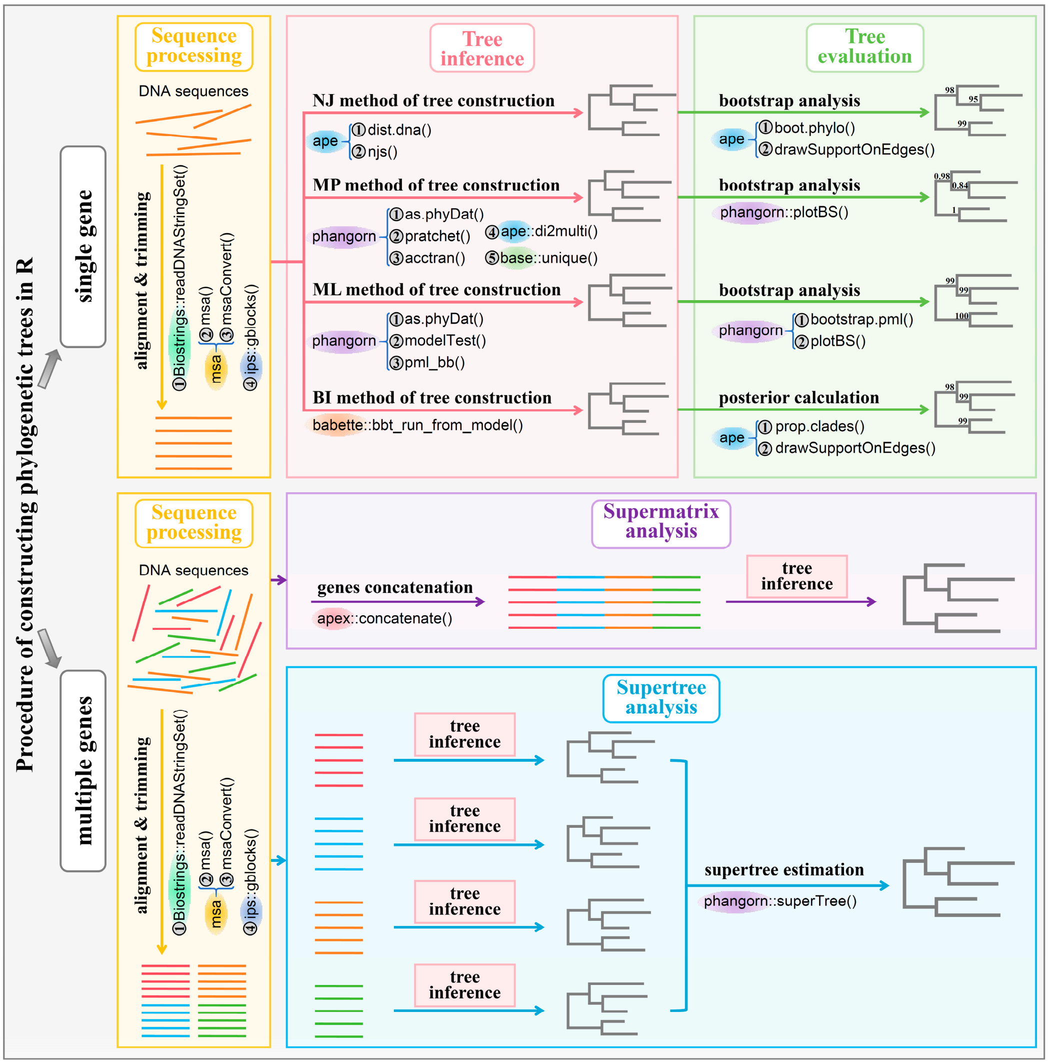


Figure 9. The whole procedure for building the phylogenetic trees in R environment. The abbreviations in the figure are as follows: NJ, neighbor-joining; MP, maximum parsimony; ML, maximum likelihood; BI, Bayesian inference.

4.1. Implementation of Distance-Based Methods in R

The NJ method can be implemented using relevant functions in the ape package. The ape package is primarily used for reading, writing, analyzing, and simulating phylogenetic trees and DNA sequences, calculating DNA distances, translating DNA sequences into protein sequences, and estimating phylogenetic trees using distance-based methods for evaluation [92]. Listing 1 exhibits the complete process of building a phylogenetic tree using the NJ method under the R programming environment. After installing and loading the ape package, the *dist.dna()* function is used to generate the distance matrix of DNA sequences. Users can select the desired molecular evolution model by setting the model parameter, with the default being the K80 model. Based on this distance matrix (which can contain missing values), the phylogenetic tree (unrooted tree) is constructed using the *njs()* function. The *root()* function is used to set *Escherichia coli* (ece) as an outgroup to define the root. To convert the phylogenetic tree properties to a rooted tree, the parameter r = TRUE needs to be set. Subsequently, *boot.phylo()* function is used for bootstrap analysis, visualized using the *plot()* function, and bootstrap values are added using the *drawSupportOnEdges()* function.

Listing 1. The code for implementation of the neighbor-joining method in R.

```
setwd(“D:/R_study/Constructing_phylogenetic_trees/”)
library(adegenet)
# reading alignments with the fasta format and outputing a DNAbin object
dna_K00927 <- fasta2DNAbin(“./data/ntK00927.fas”)
## method_1: NJ
#install.packages(“ape”)
library(ape)
# computing a matrix of pairwise distances from DNA sequences
dist <- dist.dna(dna_K00927)
# reconstructing a NJ tree from a distance matrix with possibly missing values
treenj <- njs(dist)
treenj <- root(treenj, outgroup = “ece”, r = TRUE)
# bootstrap analysis
bstrees <- boot.phylo(treenj, dna_K00927,
                      FUN = function(x) nj(dist.dna(x)),
                      trees = FALSE)
# Visualizing the NJ tree
plot(treenj)
# adding bootstrap values
drawSupportOnEdges(bstrees)
```

4.2. Implementation of MP Method in R

The R package “phangorn” allows for the estimation of phylogenetic trees and networks using maximum likelihood, maximum parsimony, distance-based, and Hadamard conjugation methods. It also provides methods for tree comparison, model selection, and visualization of phylogenetic networks [93]. Listing 2 shows the workflow of building a phylogenetic tree using the MP method under the R environment.

Listing 2. The code for implementation of the maximum parsimony method in R.

```
library(phangorn)
library(ape)
# transforming the DNAbin format into the phyDat format
phydata <- as.phyDat(dna_K00927)
# searching for the MP tree(s)
treeRatchet <- pratchet(phydata, trace = 0)
# calculating the branch length by assigning edge weights
treeRatchet <- acctran(treeRatchet, phydata)
# pruning away internal edges of length tol(default = 1e-08)
treeRatchet  <- di2multi(treeRatchet)
# returning a consensus tree
if (inherits(treeRatchet, “multiPhylo”)) {
  treeRatchet <- unique(treeRatchet) }
# ploting the consensus tree with the bootstrap values
plotBS(treeRatchet)
```

After installing and calling the phangorn package, DNA sequences are converted from DNAbin format to phydata format using the *as.phyDat()* function. In this study, the parsimony ratchet method is applied to search for maximum parsimony (MP) trees using the *pratchet()* function. The minit parameter can be set to determine the minimum number of iterations, and trace = 0 is used to prevent the current state from being written to the console, which does not affect the results obtained from this function. Subsequently, branch lengths are calculated using the *acctran()* function. As the parsimony ratchet method may produce multiple MP trees, the *unique()* function can be used to generate a consensus tree. Finally, the *plotBS()* function is used to visualize the consensus tree and add bootstrap values.

4.3. Implementation of ML Method in R

The maximum likelihood method will also be implemented using relevant functions in the phangorn package. Listing 3 shows the workflow of building a phylogenetic tree using the ML method under the R environment. After converting DNA sequences from DNAbin format to phydata format, the *modelTest()* function is used to perform model testing in search of the best model. Users can compare evolutionary models by setting the model parameter. Subsequently, the *pml_bb()* function is used to infer the maximum likelihood tree using the ML method. Finally, bootstrap analysis is performed, and the tree topology is separated from the output results for visualization.

Listing 3. The code for implementation of the maximum likelihood method in R.

```r
library(phangorn)
# transforming the DNAbin format into the phyDat format
phydata <- as.phyDat(dna_K00927)
# Comparing nucleotide substitution models
modeltest <- modelTest(phydata)
# infering a tree using maximum likelihood method
pml <- pml_bb(modeltest)
# bootstrap analysis
bstrees <- bootstrap.pml(pml)
# ploting the ML tree with the bootstrap values
plotBS(pml$tree, bstrees)
```

4.4. Implementation of BI Method in R

The R package “babette” is a popular alternative workflow for the Bayesian inference software BEAST2. It generates posterior distributions for phylogenetic trees and parameter estimates based on alignment results and inference models [104]. Listing 4 shows the workflow of building a phylogenetic tree using the BI method under the R environment. After installing and calling the babette package, Bayesian inference can be directly performed using the *bbt_run_from_model()* function. The length of the Markov chain Monte Carlo (MCMC), the number of iterations per tree sample (minimum 1000), and other settings can be adjusted using the MCMC parameter in the *create_inference_model()* function. Subsequently, the hypothesis sample with the highest posterior probability is extracted as the optimal tree, and posterior probabilities for branches are calculated, followed by visualization.

4.5. Building the Consensus Phylogenetic Tree Using Multiple Genes in R

For the aforementioned sixteen species we downloaded 10 sets of orthologous genes (K01939, K03644, K00797, K00826, K00088, K02257, K00164, K00820, K06158, and K00008), and used them as input data to construct phylogenetic trees via supermatrix and supertree methods, respectively.

To implement the supermatrix (i.e., concatenation phylogeny method) in the R environment (as shown in Listing 5), the apex package and its dependent devtools package need to be installed first. After aligning the sequences, the FASTA format of these 10 orthologous genes is converted to DNAbin format, merged into a list, and then transformed into multidna format. Subsequently, they are concatenated into a “super gene” matrix, and a phylogenetic tree is constructed using the maximum likelihood method. The core of the supermatrix method lies in concatenating different gene sequences end-to-end to form a complete sequence, which can be understood as combining the sequences of ten genes into a single gene.

Listing 4. The code for implementation of the Bayesian inference method in R.

```r
library(babette)
library(ape)
# running BEAST2 for Bayesian phylogenetic analysis
outputs <- bbt_run_from_model(“./data/ntK00927.fas”,
                           inference_model = create_inference_model(),
                           beast2_options = create_beast2_options())
# extracting the tree with the highest posterior
treeBI <- outputs$ntK00927_trees[[which.max(outputs$estimates$posterior)]]
# merging all Convergent samples into a list(multiphylo)
con.trees <- outputs$ntK00927_trees[3:10001]
# calculating the posterior of the bifurcations in the BI tree
posterior <- round(prop.clades(treeBI, con.trees)/9999*100)
# Visualizing the BI tree
plot(treeBI)
# adding posterior values of the bifurcations
drawSupportOnEdges(posterior)
```

Listing 5. The code for implementation of concatenation phylogeny method in R.

```r
library(devtools)
#devtools::install_github(“thibautjombart/apex”)
library(apex)
# storing these ten genes in the same list
dna <- c(“ntK01939”, “ntK03644”, “ntK00797”, “ntK00826”, “ntK00088”,
        “ntK02257”, “ntK00164”, “ntK00820”, “ntK06158”, “ntK00008”)
i <- 0
DNAbin_list <- list()
for (k in dna) {
  i <- i + 1
  dna <- fasta2DNAbin(paste0(“./data/”, k, “.fas”))
  DNAbin_list[[i]] <- dna
}
# transforming the list of DNAbins into the multidna format
multidna <- new(“multidna”, DNAbin_list)
# concatenating the DNA alignments into a single alignement matrix
phydata <- as.phyDat(concatenate(multidna))
# Comparing nucleotide substitution models
modeltest <- modelTest(phydata)
# infering a tree using maximum likelihood method
pml <- pml_bb(modeltest)
# Visualizing the ML tree
plot(pml$tree)
```

The supertree (i.e., coalescence phylogeny method) is implemented using the phangorn package, with the *superTree()* function being the core function for constructing the supertree. After constructing all individual gene trees based on the maximum likelihood method, they are merged into a multiPhylo format. Then the *superTree()* function is used to integrate them into a single supertree using the MRP method. The whole procedure was illustrated in Listing 6. The core of the supertree method lies in integrating different individual gene trees into a single phylogenetic tree.

Listing 6. The code for implementation of coalescence phylogeny method in R.

```
library(phangorn)
# constructing sigle gene trees, storing which in a list(multiphylo)
dna <- c(“ntK01939”, “ntK03644”, “ntK00797”, “ntK00826”, “ntK00088”,
        “ntK02257”, “ntK00164”, “ntK00820”, “ntK06158”, “ntK00008”)
i <- 0
mtree <- c()
for (k in dna) {
  i <- i + 1
  dna <- fasta2DNAbin(paste0(“./data/”, k, “.fas”))
  # transforming the DNAbin format into the phyDat format
  phydata <- as.phyDat(dna)
  # Comparing nucleotide substitution models
  modeltest <- modelTest(phydata)
  # infering a tree using maximum likelihood method
  pml <- pml_bb(modeltest)
  # storing the ML tree into the list
  mtree[[i]] <- pml$tree
}
class(mtree) <- “multiPhylo”
# estimating a supertree from a set of trees
supertree <- superTree(mtree)
# Visualizing the supertree
plot(supertree)
```

5. Summary and Perspectives

Developing methods for constructing phylogenetic trees for different purposes (such as theoretical method innovation and faster computing speed) is an important research field. Phylogenetic trees can intuitively reflect the evolutionary history and relationships among taxonomic units, helping us understand biodiversity and evolutionary patterns. R language is a very powerful statistical analysis and plotting tool, providing many packages for constructing and analyzing phylogenetic trees. Table 2 lists the common R packages used in phylogenetic tree construction. This article reviews the methods for constructing phylogenetic trees and their implementation in the R environment. Through discussions of distance methods, maximum parsimony, maximum likelihood, Bayesian inference, and phylogenetic tree integration methods, and comprehensive analysis of the advantages, disadvantages, and applicable scenarios of different methods, this article provides a reference basis for researchers to choose appropriate methods. Meanwhile, by demonstrating how to use R packages and algorithms to construct phylogenetic trees and providing code examples and practical cases, readers can better understand and apply the knowledge learned.

It is worth noting that with the rapid increase in high-throughput data such as genomics and proteomics, the methods for constructing phylogenetic trees also need to be constantly updated and improved. Currently, new methods are constantly emerging, providing research directions for phylogenetic analysis. Fusang, proposed by Wang et al. [22], is a framework for reconstructing the topology of phylogenetic trees (without calculating branch lengths) through deep learning methods, aiming to provide an evolvable toolkit for daily phylogenetic tree inference applications. However, when there are too many species (>40), the computational efficiency of the current version of Fusang is low, and it currently only supports amino acid multisequence alignments with fewer than 40 sequences and sequence lengths below 10,000. Fusang uses an improved stepwise addition algorithm inspired by Zou et al. [105] to solve variable MSA sequences and uses beam search to find the best topology based on the probability distribution provided by deep learning. When there are more insertions and deletions in the multiple sequence alignment, DL performs better than ML methods because DL adds indel information to greatly improve the accuracy of phylogenetic inference. The authors have also pushed related Python code on GitHub for users to use Fusang.

Table 2. The common R packages used in phylogenetic tree construction.

R Package	Description	Source	Reference
ape	Providing both utility functions for reading and writing data and manipulating phylogenetic trees, as well as several advanced methods for phylogenetic and evolutionary analysis.	CRAN *	[92]
phangorn	Estimating phylogenetic trees and networks using maximum likelihood, maximum parsimony, distance methods, and Hadamard conjugation; offering methods for tree comparison, model selection, and visualization of phylogenetic networks.	CRAN *	[93]
babette	Providing an alternative workflow to the BEAST2; conducting complex Bayesian phylogenetics easily and reproducibly from R.	Github	[106]
BAMMtools	Reconstructing and visualizing changes in evolutionary rates through time and across clades in a Bayesian statistical framework.	CRAN *	[107]
apex	Implementing new object classes for storing and handling multiple genes data.	CRAN *	[108]
phytools	Concentrating on phylogenetic comparative biology; including numerous techniques for visualizing, analyzing, manipulating, reading or writing, and inferring phylogenetic trees.	CRAN *	[109]
ggtree	Annotating phylogenetic trees with their associated data of different types and from various sources.	Bioconductor	[97]
RPANDA	Characterizing and comparing phylogenies using spectral densities; fitting models of diversification to phylogenies.	CRAN *	[110]
TreeSearch	Dataset construction and validation; phylogenetic search (including with inapplicable data); the interrogation of optimal tree sets.	CRAN *	[111]
paleotree	Analyzing the combined paleontological and phylogenetic data sets, particularly the time-scaling of phylogenetic trees, which include extinct fossil lineages.	CRAN *	[112]
treeman	Containing a new class called TreeMan for representing phylogenetic trees that has a list structure that allows for more efficient manipulation of phylogenetic trees; demonstrating intuitive tree manipulation, both conceptually and as computationally efficient as possible, within the R environment.	Github	[113]

* CRAN: The Comprehensive R Archive Network.

ModelRevelator, proposed by Burgstaller-Muehlbacher et al. [114], is a machine learning method supported by two neural networks (deep learning) for model selection in phylogenetic inference, which aims to find the best model for sequence evolution and tends to choose models with fewer parameters. The authors demonstrated that neural networks can be used for model selection without rebuilding trees, optimizing parameters, or calculating likelihoods. The first neural network, NNmodelfind, recommends one of six commonly used sequence evolution models (JC, K2P, F81, HKY, TN93, and GTR). The second neural network, NNalphafind, provides an estimate of the shape parameter α by suggesting whether to merge gamma-distributed rate heterogeneity models. The authors found that using neural networks for phylogenetic estimation slightly improved compared to ML + BIC and significantly saved computing time depending on the size of the alignment. Users can use ModelRevelator for phylogenetic analysis through IQ-tree software.

Hyperbolic embedding in phylogenetic analysis embeds the sequences of taxonomic groups into hyperbolic spaces using hyperbolic geometry models (the hyperboloid model, Klein disk model, and Poincaré disk model), represented as points, and calculates the distances between them. Hyperbolic space has negative curvature (negative curvature: the sum of interior angles of any triangle on the surface is less than π), and its exponential expansion rate is much greater than that of Euclidean space. Therefore, compared with Euclidean embedding, hyperbolic embedding more closely matches the geometric shape of trees [115] and better represents hierarchical structures [116]. However, hyperbolic embedding is currently significantly more effective than Euclidean embedding only in low dimensions and loses its advantage in high dimensions [117]. Macaulay et al. [118] applied hyperbolic embedding to Bayesian phylogenetic analysis, studying the impact of the curvature (the degree of geometric curvature) and dimension of hyperbolic space on MCMC chain performance, and concluded that hyperbolic embedding allows tree search algorithms to propose new states (topology and branch length) from continuous probability distributions.

New methods of phylogenetic analysis offer distinct advantages over traditional methods but also present challenges that require further refinement by researchers. Future investigations should prioritize gaining a comprehensive understanding of current methods and techniques while critically assessing their limitations, thus providing an essential context for the integration of innovative approaches. Meeting the growing demand for large data sets requires the development of more efficient and accurate methods, as well as deeper investigations into the integration of artificial intelligence and machine learning technologies into phylogenetic tree construction. Encouragingly, the use of advanced large language models (LLMs) [119], such as OpenAI's ChatGPT [120], known for its exceptional language processing and programming capabilities, offers promising prospects for advancing phylogenetic research. Continued optimization of existing phylogenetic analysis methods and exploration of new techniques within the R programming environment will enable researchers to harness large amounts of data for iterative analysis, resulting in the construction of more robust and comprehensive phylogenetic trees that accurately reflect the evolutionary relationships between species. In addition, this approach may facilitate the identification of minimal orthologous gene sets with whole genome representation.

Author Contributions: Conceptualization, Y.Z. (Yue Zou) and Z.Z.; methodology, Y.Z. (Yue Zou) and B.L.; data curation, Y.Z. (Yue Zou), Y.Z. (Yujie Zeng) and H.H.; writing—original draft preparation, Y.Z. (Yue Zou); writing—review and editing, Z.Z.; figure generation and visualization, Y.Z. (Yue Zou) and Y.H.; supervision, S.H. and B.L.; project administration, B.L. and S.H.; funding acquisition, S.H. and B.L. All authors have read and agreed to the published version of the manuscript.

Funding: This research was sponsored by the Chongqing Higher Education Teaching Reform Research Project (No. 233183), the Science and Technology Research Program of the Chongqing Municipal Education Commission (No. KJQN202100538), the Talent Innovation Project in Life Sciences of Chongqing Normal University (No. CSSK2023-04), and the Open Fund of the Yunnan Key Laboratory of Plant Reproductive Adaptation and Evolutionary Ecology, Yunnan University (YNPRAEC-2023004).

Data Availability Statement: All R codes and datasets used in this review are freely available at https://github.com/libcell/phylosinR (accessed on 4 April 2024).

Acknowledgments: We appreciate the assistance provided by Dan Qin (Department of Biology, College of Science, Northeastern University, Boston, USA) in revising the manuscript and polishing the English language.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Sanderson, M.J.; Driskell, A.C. The challenge of constructing large phylogenetic trees. *Trends Plant Sci.* **2003**, *8*, 374–379. [CrossRef] [PubMed]
2. Hug, L.A.; Baker, B.J.; Anantharaman, K.; Brown, C.T.; Probst, A.J.; Castelle, C.J.; Butterfield, C.N.; Hernsdorf, A.W.; Amano, Y.; Ise, K.; et al. A new view of the tree of life. *Nat. Microbiol.* **2016**, *1*, 16048. [CrossRef] [PubMed]
3. Abaza, S. What is and why do we have to know the phylogenetic tree? *Parasitol. United J.* **2020**, *13*, 68–71. [CrossRef]
4. de Queiroz, K. Nodes, branches, and phylogenetic definitions. *Syst. Biol.* **2013**, *62*, 625–632. [CrossRef] [PubMed]
5. Dissanayake, A.; Bhunjun, C.; Maharachchikumbura, S.; Liu, J. Applied aspects of methods to infer phylogenetic relationships amongst fungi. *Mycosphere* **2020**, *11*, 2652–2676. [CrossRef]
6. Gupta, M.K.; Gouda, G.; Sabarinathan, S.; Donde, R.; Rajesh, N.; Pati, P.; Rathore, S.K.; Behera, L.; Vadde, R. Phylogenetic analysis. In *Bioinformatics in Rice Research: Theories and Techniques*; Springer: Singapore, 2021; pp. 179–207.
7. Feng, H.; Liu, M.; Wang, B.; Feng, J.; Han, J.; Liu, J. HCPC: A New Parsimonious Clustering Method based on Hierarchical Characters for Morphological Phylogenetic Reconstruction. *Res. Sq.* **2021**. [CrossRef]
8. Mc, C.E.; Verdeflor, L.; Weinsztok, A.; Wiles, J.R.; Dorus, S. Exploratory Activities for Understanding Evolutionary Relationships Depicted by Phylogenetic Trees: United but Diverse. *Am. Biol. Teach.* **2020**, *82*, 333–337. [CrossRef]
9. Jetz, W.; Thomas, G.H.; Joy, J.B.; Hartmann, K.; Mooers, A.O. The global diversity of birds in space and time. *Nature* **2012**, *491*, 444–448. [CrossRef]
10. Hinchliff, C.E.; Smith, S.A.; Allman, J.F.; Burleigh, J.G.; Chaudhary, R.; Coghill, L.M.; Crandall, K.A.; Deng, J.; Drew, B.T.; Gazis, R.; et al. Synthesis of phylogeny and taxonomy into a comprehensive tree of life. *Proc. Natl. Acad. Sci. USA* **2015**, *112*, 12764–12769. [CrossRef]
11. Denamur, E.; Clermont, O.; Bonacorsi, S.; Gordon, D. The population genetics of pathogenic Escherichia coli. *Nat. Rev. Microbiol.* **2021**, *19*, 37–54. [CrossRef]
12. Smith, S.D.; Pennell, M.W.; Dunn, C.W.; Edwards, S.V. Phylogenetics is the New Genetics (for Most of Biodiversity). *Trends Ecol. Evol.* **2020**, *35*, 415–425. [CrossRef]
13. Lee, M.S.; Palci, A. Morphological Phylogenetics in the Genomic Age. *Curr. Biol. CB* **2015**, *25*, R922–R929. [CrossRef] [PubMed]
14. Lemmon, E.M.; Lemmon, A.R. High-throughput genomic data in systematics and phylogenetics. *Annu. Rev. Ecol. Evol. Syst.* **2013**, *44*, 99–121. [CrossRef]
15. Morel, B.; Williams, T.A.; Stamatakis, A. Asteroid: A new algorithm to infer species trees from gene trees under high proportions of missing data. *Bioinformatics* **2023**, *39*, btac832. [CrossRef]
16. James, T.Y.; Stajich, J.E.; Hittinger, C.T.; Rokas, A. Toward a Fully Resolved Fungal Tree of Life. *Annu. Rev. Microbiol.* **2020**, *74*, 291–313. [CrossRef] [PubMed]
17. Ashkenazy, H.; Sela, I.; Levy Karin, E.; Landan, G.; Pupko, T. Multiple Sequence Alignment Averaging Improves Phylogeny Reconstruction. *Syst. Biol.* **2019**, *68*, 117–130. [CrossRef]
18. Francis, W.R.; Canfield, D.E. Very few sites can reshape the inferred phylogenetic tree. *PeerJ* **2020**, *8*, e8865. [CrossRef] [PubMed]
19. Talavera, G.; Castresana, J. Improvement of phylogenies after removing divergent and ambiguously aligned blocks from protein sequence alignments. *Syst. Biol.* **2007**, *56*, 564–577. [CrossRef] [PubMed]
20. Williams, T.A.; Heaps, S.E. An introduction to phylogenetics and the tree of life. In *Methods in Microbiology*; Elsevier: Amsterdam, The Netherlands, 2014; Volume 41, pp. 13–44.
21. Desper, R.; Gascuel, O. Theoretical foundation of the balanced minimum evolution method of phylogenetic inference and its relationship to weighted least-squares tree fitting. *Mol. Biol. Evol.* **2004**, *21*, 587–598. [CrossRef]
22. Wang, Z.; Sun, J.; Gao, Y.; Xue, Y.; Zhang, Y.; Li, K.; Zhang, W.; Zhang, C.; Zu, J.; Zhang, L. Fusang: A framework for phylogenetic tree inference via deep learning. *Nucleic Acids Res.* **2023**, *51*, 10909–10923. [CrossRef]
23. Balaban, M.; Jiang, Y.; Roush, D.; Zhu, Q.; Mirarab, S. Fast and accurate distance-based phylogenetic placement using divide and conquer. *Mol. Ecol. Resour.* **2022**, *22*, 1213–1227. [CrossRef] [PubMed]
24. Vaz, C.; Nascimento, M.; Carriço, J.A.; Rocher, T.; Francisco, A.P. Distance-based phylogenetic inference from typing data: A unifying view. *Brief. Bioinform.* **2021**, *22*, bbaa147. [CrossRef] [PubMed]
25. Coorens, T.H.; Spencer Chapman, M.; Williams, N.; Martincorena, I.; Stratton, M.R.; Nangalia, J.; Campbell, P.J. Reconstructing phylogenetic trees from genome-wide somatic mutations in clonal samples. *Nat. Protoc.* **2024**, 1–21. [CrossRef] [PubMed]
26. Scossa, F.; Fernie, A.R. Ancestral sequence reconstruction—An underused approach to understand the evolution of gene function in plants? *Comput. Struct. Biotechnol. J.* **2021**, *19*, 1579–1594. [CrossRef] [PubMed]

27. Ojha, K.K.; Mishra, S.; Singh, V.K. Computational molecular phylogeny: Concepts and applications. In *Bioinformatics*; Academic Press: New York, NY, USA, 2022; pp. 67–89.
28. Kapli, P.; Yang, Z.; Telford, M.J. Phylogenetic tree building in the genomic age. *Nat. Rev. Genet.* **2020**, *21*, 428–444. [CrossRef] [PubMed]
29. Mount, D.W. Distance methods for phylogenetic prediction. *CSH Protoc.* **2008**, *2008*, pdb.top33. [CrossRef] [PubMed]
30. Davidson, R.; Martín Del Campo, A. Combinatorial and Computational Investigations of Neighbor-Joining Bias. *Front. Genet.* **2020**, *11*, 584785. [CrossRef] [PubMed]
31. Saitou, N.; Nei, M. The neighbor-joining method: A new method for reconstructing phylogenetic trees. *Mol. Biol. Evol.* **1987**, *4*, 406–425. [CrossRef] [PubMed]
32. Kuhner, M.K.; Felsenstein, J. A simulation comparison of phylogeny algorithms under equal and unequal evolutionary rates. *Mol. Biol. Evol.* **1994**, *11*, 459–468. [CrossRef]
33. Godini, R.; Fallahi, H. A brief overview of the concepts, methods and computational tools used in phylogenetic tree construction and gene prediction. *Meta Gene* **2019**, *21*, 100586. [CrossRef]
34. Tamura, K.; Nei, M.; Kumar, S. Prospects for inferring very large phylogenies by using the neighbor-joining method. *Proc. Natl. Acad. Sci. USA* **2004**, *101*, 11030–11035. [CrossRef]
35. Zhang, L.-N.; Rong, C.-H.; He, Y.; Guan, Q.; He, B.; Zhu, X.-W.; Liu, J.-N.; Chen, H.-J. A bird's eye view of the algorithms and software packages for reconstructing phylogenetic trees. *Zool. Res.* **2013**, *34*, 640–650. [CrossRef]
36. Santiago-Alarcon, D.; Tapia-McClung, H.; Lerma-Hernández, S.; Venegas-Andraca, S.E. Quantum aspects of evolution: A contribution towards evolutionary explorations of genotype networks via quantum walks. *J. R. Soc. Interface* **2020**, *17*, 20200567. [CrossRef]
37. Farris, J.S. Methods for computing Wagner trees. *Syst. Biol.* **1970**, *19*, 83–92. [CrossRef]
38. Fitch, W.M. Toward defining the course of evolution: Minimum change for a specific tree topology. *Syst. Biol.* **1971**, *20*, 406–416. [CrossRef]
39. Liu, D.K.; Tu, X.D.; Zhao, Z.; Zeng, M.Y.; Zhang, S.; Ma, L.; Zhang, G.Q.; Wang, M.M.; Liu, Z.J.; Lan, S.R.; et al. Plastid phylogenomic data yield new and robust insights into the phylogeny of Cleisostoma-Gastrochilus clades (Orchidaceae, Aeridinae). *Mol. Phylogenetics Evol.* **2020**, *145*, 106729. [CrossRef] [PubMed]
40. Azouri, D.; Abadi, S.; Mansour, Y.; Mayrose, I.; Pupko, T. Harnessing machine learning to guide phylogenetic-tree search algorithms. *Nat. Commun.* **2021**, *12*, 1983. [CrossRef] [PubMed]
41. Felsenstein, J. Evolutionary trees from DNA sequences: A maximum likelihood approach. *J. Mol. Evol.* **1981**, *17*, 368–376. [CrossRef]
42. Jukes, T.H.; Cantor, C.R. Evolution of protein molecules. *Mamm. Protein Metab.* **1969**, *3*, 21–132.
43. Wascher, M.; Kubatko, L. Consistency of SVDQuartets and Maximum Likelihood for Coalescent-Based Species Tree Estimation. *Syst. Biol.* **2021**, *70*, 33–48. [CrossRef]
44. Kimura, M. A simple method for estimating evolutionary rates of base substitutions through comparative studies of nucleotide sequences. *J. Mol. Evol.* **1980**, *16*, 111–120. [CrossRef] [PubMed]
45. Tamura, K.; Nei, M. Estimation of the number of nucleotide substitutions in the control region of mitochondrial DNA in humans and chimpanzees. *Mol. Biol. Evol.* **1993**, *10*, 512–526. [CrossRef] [PubMed]
46. Hasegawa, M.; Kishino, H.; Yano, T. Dating of the human-ape splitting by a molecular clock of mitochondrial DNA. *J. Mol. Evol.* **1985**, *22*, 160–174. [CrossRef] [PubMed]
47. Tavaré, S. Some probabilistic and statistical problems on the analysis of DNA sequence. *Lect. Math. Life Sci.* **1986**, *17*, 57.
48. Jacob, S.S.; Sengupta, P.P.; Chandu, A.G.S.; Shamshad, S.; Yogisharadhya, R.; Sudhagar, S.; Ramesh, P. Existence of genetic lineages within Asian genotype of Taenia solium-Genetic characterization based on mitochondrial and ribosomal DNA markers. *Transbound. Emerg. Dis.* **2022**, *69*, 2256–2265. [CrossRef]
49. Heaps, S.E.; Nye, T.M.; Boys, R.J.; Williams, T.A.; Embley, T.M. Bayesian modelling of compositional heterogeneity in molecular phylogenetics. *Stat. Appl. Genet. Mol. Biol.* **2014**, *13*, 589–609. [CrossRef] [PubMed]
50. Amiroch, S.; Pradana, M.S.; Irawan, M.I.; Mukhlash, I. Maximum Likelihood Method on The Construction of Phylogenetic Tree for Identification the Spreading of SARS Epidemic. In Proceedings of the 2018 International Symposium on Advanced Intelligent Informatics (SAIN), Yogyakarta, Indonesia, 29–30 August 2018; pp. 137–141.
51. Rannala, B.; Yang, Z. Probability distribution of molecular evolutionary trees: A new method of phylogenetic inference. *J. Mol. Evol.* **1996**, *43*, 304–311. [CrossRef] [PubMed]
52. Flouri, T.; Huang, J.; Jiao, X.; Kapli, P.; Rannala, B.; Yang, Z. Bayesian Phylogenetic Inference using Relaxed-clocks and the Multispecies Coalescent. *Mol. Biol. Evol.* **2022**, *39*, msac161. [CrossRef] [PubMed]
53. Nascimento, F.F.; Reis, M.D.; Yang, Z. A biologist's guide to Bayesian phylogenetic analysis. *Nat. Ecol. Evol.* **2017**, *1*, 1446–1454. [CrossRef]
54. Cornuault, J.; Sanmartín, I. A road map for phylogenetic models of species trees. *Mol. Phylogenetics Evol.* **2022**, *173*, 107483. [CrossRef]
55. Spade, D.A. Geometric ergodicity of a Metropolis-Hastings algorithm for Bayesian inference of phylogenetic branch lengths. *Comput. Stat.* **2020**, *35*, 2043–2076. [CrossRef]

56. Csősz, S.; Loss, A.C.; Fisher, B.L. Exploring the diversity of the Malagasy Ponera (Hymenoptera: Formicidae) fauna via integrative taxonomy. *Org. Divers. Evol.* **2023**, *23*, 917–927. [CrossRef]
57. Larget, B.; Simon, D.L. Markov Chasin Monte Carlo Algorithms for the Bayesian Analysis of Phylogenetic Trees. *Mol. Biol. Evol.* **1999**, *16*, 750. [CrossRef]
58. Whidden, C.; Matsen, F.A.t. Quantifying MCMC exploration of phylogenetic tree space. *Syst. Biol.* **2015**, *64*, 472–491. [CrossRef]
59. Inagaki, Y.; Nakajima, Y.; Sato, M.; Sakaguchi, M.; Hashimoto, T. Gene sampling can bias multi-gene phylogenetic inferences: The relationship between red algae and green plants as a case study. *Mol. Biol. Evol.* **2009**, *26*, 1171–1178. [CrossRef]
60. Lax, G.; Kolisko, M.; Eglit, Y.; Lee, W.J.; Yubuki, N.; Karnkowska, A.; Leander, B.S.; Burger, G.; Keeling, P.J.; Simpson, A.G.B. Multigene phylogenetics of euglenids based on single-cell transcriptomics of diverse phagotrophs. *Mol. Phylogenetics Evol.* **2021**, *159*, 107088. [CrossRef]
61. Kanzi, A.M.; Trollip, C.; Wingfield, M.J.; Barnes, I.; Van der Nest, M.A.; Wingfield, B.D. Phylogenomic incongruence in Ceratocystis: A clue to speciation? *BMC Genom.* **2020**, *21*, 362. [CrossRef] [PubMed]
62. Williams, T.A.; Cox, C.J.; Foster, P.G.; Szöllősi, G.J.; Embley, T.M. Phylogenomics provides robust support for a two-domains tree of life. *Nat. Ecol. Evol.* **2020**, *4*, 138–147. [CrossRef]
63. Pardo-De la Hoz, C.J.; Magain, N.; Piatkowski, B.; Cornet, L.; Dal Forno, M.; Carbone, I.; Miadlikowska, J.; Lutzoni, F. Ancient Rapid Radiation Explains Most Conflicts Among Gene Trees and Well-Supported Phylogenomic Trees of Nostocalean Cyanobacteria. *Syst. Biol.* **2023**, *72*, 694–712. [CrossRef]
64. Shen, X.X.; Li, Y.; Hittinger, C.T.; Chen, X.X.; Rokas, A. An investigation of irreproducibility in maximum likelihood phylogenetic inference. *Nat. Commun.* **2020**, *11*, 6096. [CrossRef]
65. Zhao, P.; Kakishima, M.; Uzuhashi, S.; Ishii, H. Multigene phylogenetic analysis of inter- and intraspecific relationships in Venturia nashicola and V. pirina. *Eur. J. Plant Pathol.* **2012**, *132*, 245–258. [CrossRef]
66. Abeysundera, M.; Field, C.; Gu, H. Phylogenetic Analysis Based on Spectral Methods. *Mol. Biol. Evol.* **2012**, *29*, 579–597. [CrossRef]
67. Bi, G.; Mao, Y.; Xing, Q.; Cao, M. HomBlocks: A multiple-alignment construction pipeline for organelle phylogenomics based on locally collinear block searching. *Genomics* **2018**, *110*, 18–22. [CrossRef]
68. Steenwyk, J.L.; Li, Y.; Zhou, X.; Shen, X.X.; Rokas, A. Incongruence in the phylogenomics era. *Nat. Rev. Genet.* **2023**, *24*, 834–850. [CrossRef] [PubMed]
69. Wolsan, M.; Sato, J.J. Effects of data incompleteness on the relative performance of parsimony and Bayesian approaches in a supermatrix phylogenetic reconstruction of Mustelidae and Procyonidae (Carnivora). *Cladistics Int. J. Willi Hennig Soc.* **2010**, *26*, 168–194. [CrossRef]
70. Rannala, B.; Yang, Z. Phylogenetic inference using whole genomes. *Annu. Rev. Genom. Hum. Genet.* **2008**, *9*, 217–231. [CrossRef] [PubMed]
71. Zou, X.-H.; Song, G. Conflicting gene trees and phylogenomics. *J. Syst. Evol.* **2008**, *46*, 795.
72. Delsuc, F.; Brinkmann, H.; Philippe, H. Phylogenomics and the reconstruction of the tree of life. *Nat. Rev. Genet.* **2005**, *6*, 361–375. [CrossRef]
73. Bininda-Emonds, O.R.; Sanderson, M.J. Assessment of the accuracy of matrix representation with parsimony analysis supertree construction. *Syst. Biol.* **2001**, *50*, 565–579. [CrossRef]
74. Zhao, T.; Zwaenepoel, A.; Xue, J.-Y.; Kao, S.-M.; Li, Z.; Schranz, M.E.; Van de Peer, Y. Whole-genome microsynteny-based phylogeny of angiosperms. *Nat. Commun.* **2021**, *12*, 3498. [CrossRef]
75. Cotton, J.A.; Wilkinson, M. Majority-rule supertrees. *Syst. Biol.* **2007**, *56*, 445–452. [CrossRef]
76. Delucchi, E.; Hoessly, L.; Paolini, G. Impossibility Results on Stability of Phylogenetic Consensus Methods. *Syst. Biol.* **2020**, *69*, 557–565. [CrossRef] [PubMed]
77. Goloboff, P.A.; Pol, D. Semi-strict supertrees. *Cladistics Int. J. Willi Hennig Soc.* **2002**, *18*, 514–525. [CrossRef]
78. Fischer, M.; Hendriksen, M. Refinement-stable Consensus Methods. *arXiv* **2021**, arXiv:2102.04502. [CrossRef]
79. Lapointe, F.-J.; Cucumel, G. The Average Consensus Procedure: Combination of Weighted Trees Containing Identical or Overlapping Sets of Taxa. *Syst. Biol.* **1997**, *46*, 306–312. [CrossRef]
80. Mavrodiev, E.V.; Williams, D.M.; Ebach, M.C. On the Typology of Relations. *Evol. Biol.* **2019**, *46*, 71–89. [CrossRef]
81. Lu, L.; Sun, M.; Zhang, J.; Li, H.; Lin, L.; Yang, T.; Chen, M.; Chen, Z. Tree of life and its applications. *Biodivers. Sci.* **2014**, *22*, 3–20. [CrossRef]
82. Jiang, X.; Edwards, S.V.; Liu, L. The Multispecies Coalescent Model Outperforms Concatenation Across Diverse Phylogenomic Data Sets. *Syst. Biol.* **2020**, *69*, 795–812. [CrossRef]
83. Retief, J.D. Phylogenetic analysis using PHYLIP. *Methods Mol. Biol.* **2000**, *132*, 243–258. [CrossRef]
84. Wilgenbusch, J.C.; Swofford, D. Inferring evolutionary trees with PAUP*. In *Current Protocols in Bioinformatics*; Wiley: Hoboken, NJ, USA, 2003; Chapter 6, Unit 6.4. [CrossRef]
85. Guindon, S.; Dufayard, J.F.; Lefort, V.; Anisimova, M.; Hordijk, W.; Gascuel, O. New algorithms and methods to estimate maximum-likelihood phylogenies: Assessing the performance of PhyML 3.0. *Syst. Biol.* **2010**, *59*, 307–321. [CrossRef]
86. Ronquist, F.; Teslenko, M.; van der Mark, P.; Ayres, D.L.; Darling, A.; Höhna, S.; Larget, B.; Liu, L.; Suchard, M.A.; Huelsenbeck, J.P. MrBayes 3.2: Efficient Bayesian phylogenetic inference and model choice across a large model space. *Syst. Biol.* **2012**, *61*, 539–542. [CrossRef]

87. Tamura, K.; Stecher, G.; Kumar, S. MEGA11: Molecular Evolutionary Genetics Analysis Version 11. *Mol. Biol. Evol.* **2021**, *38*, 3022–3027. [CrossRef] [PubMed]
88. Xiang, C.Y.; Gao, F.; Jakovlić, I.; Lei, H.P.; Hu, Y.; Zhang, H.; Zou, H.; Wang, G.T.; Zhang, D. Using PhyloSuite for molecular phylogeny and tree-based analyses. *iMeta* **2023**, *2*, e87. [CrossRef]
89. Huber, W.; Carey, V.J.; Gentleman, R.; Anders, S.; Carlson, M.; Carvalho, B.S.; Bravo, H.C.; Davis, S.; Gatto, L.; Girke, T.; et al. Orchestrating high-throughput genomic analysis with Bioconductor. *Nat. Methods* **2015**, *12*, 115–121. [CrossRef] [PubMed]
90. Giorgi, F.M.; Ceraolo, C.; Mercatelli, D. The R Language: An Engine for Bioinformatics and Data Science. *Life* **2022**, *12*, 648. [CrossRef] [PubMed]
91. Yu, G.; Smith, D.K.; Zhu, H.; Guan, Y.; Lam, T.T.Y. ggtree: An R package for visualization and annotation of phylogenetic trees with their covariates and other associated data. *Methods Ecol. Evol.* **2017**, *8*, 28–36. [CrossRef]
92. Paradis, E.; Schliep, K. ape 5.0: An environment for modern phylogenetics and evolutionary analyses in R. *Bioinformatics* **2019**, *35*, 526–528. [CrossRef] [PubMed]
93. Schliep, K.P. phangorn: Phylogenetic analysis in R. *Bioinformatics* **2011**, *27*, 592–593. [CrossRef] [PubMed]
94. Galili, T. dendextend: An R package for visualizing, adjusting and comparing trees of hierarchical clustering. *Bioinformatics* **2015**, *31*, 3718–3720. [CrossRef]
95. Wang, L.G.; Lam, T.T.; Xu, S.; Dai, Z.; Zhou, L.; Feng, T.; Guo, P.; Dunn, C.W.; Jones, B.R.; Bradley, T.; et al. Treeio: An R Package for Phylogenetic Tree Input and Output with Richly Annotated and Associated Data. *Mol. Biol. Evol.* **2020**, *37*, 599–603. [CrossRef]
96. Yu, G. *Data Integration, Manipulation and Visualization of Phylogenetic Trees*; Chapman and Hall/CRC: Boca Raton, FL, USA, 2022.
97. Xu, S.; Li, L.; Luo, X.; Chen, M.; Tang, W.; Zhan, L.; Dai, Z.; Lam, T.T.; Guan, Y.; Yu, G. Ggtree: A serialized data object for visualization of a phylogenetic tree and annotation data. *iMeta* **2022**, *1*, e56. [CrossRef]
98. Wilkinson, L. ggplot2: Elegant Graphics for Data Analysis by WICKHAM, H. *Biometrics* **2011**, *67*, 678–679. [CrossRef]
99. Cock, P.J.; Antao, T.; Chang, J.T.; Chapman, B.A.; Cox, C.J.; Dalke, A.; Friedberg, I.; Hamelryck, T.; Kauff, F.; Wilczynski, B.; et al. Biopython: Freely available Python tools for computational molecular biology and bioinformatics. *Bioinformatics* **2009**, *25*, 1422–1423. [CrossRef]
100. Sukumaran, J.; Holder, M.T. DendroPy: A Python library for phylogenetic computing. *Bioinformatics* **2010**, *26*, 1569–1571. [CrossRef] [PubMed]
101. Hao, J.; Ho, T.K. Machine learning made easy: A review of scikit-learn package in python programming language. *J. Educ. Behav. Stat.* **2019**, *44*, 348–361. [CrossRef]
102. Ketkar, N.; Moolayil, J.; Ketkar, N.; Moolayil, J. Introduction to pytorch. In *Deep Learning with Python: Learn Best Practices of Deep Learning Models with PyTorch*; Apress: Berkeley, CA, USA, 2021; pp. 27–91.
103. Jombart, T. adegenet: A R package for the multivariate analysis of genetic markers. *Bioinformatics* **2008**, *24*, 1403–1405. [CrossRef]
104. Bilderbeek, R.J.; Laudanno, G.; Etienne, R.S. Quantifying the impact of an inference model in Bayesian phylogenetics. *Methods Ecol. Evol.* **2021**, *12*, 351–358. [CrossRef]
105. Zou, Z.; Zhang, H.; Guan, Y.; Zhang, J. Deep Residual Neural Networks Resolve Quartet Molecular Phylogenies. *Mol. Biol. Evol.* **2020**, *37*, 1495–1507. [CrossRef] [PubMed]
106. Bilderbeek, R.J.C.; Etienne, R.S. babette: BEAUti 2, BEAST2 and Tracer for R. *Methods Ecol. Evol.* **2018**, *9*, 2034–2040. [CrossRef]
107. Rabosky, D.L.; Grundler, M.; Anderson, C.; Title, P.; Shi, J.J.; Brown, J.W.; Huang, H.; Larson, J.G. BAMMtools: An R package for the analysis of evolutionary dynamics on phylogenetic trees. *Methods Ecol. Evol.* **2014**, *5*, 701–707. [CrossRef]
108. Jombart, T.; Archer, F.; Schliep, K.; Kamvar, Z.; Harris, R.; Paradis, E.; Goudet, J.; Lapp, H. apex: Phylogenetics with multiple genes. *Mol. Ecol. Resour.* **2017**, *17*, 19–26. [CrossRef]
109. Revell, L.J. phytools 2.0: An updated R ecosystem for phylogenetic comparative methods (and other things). *PeerJ* **2024**, *12*, e16505. [CrossRef]
110. Morlon, H.; Lewitus, E.; Condamine, F.L.; Manceau, M.; Clavel, J.; Drury, J. RPANDA: An R package for macroevolutionary analyses on phylogenetic trees. *Methods Ecol. Evol.* **2016**, *7*, 589–597. [CrossRef]
111. Smith, M.R. TreeSearch: Morphological phylogenetic analysis in R. *bioRxiv* **2021**. [CrossRef]
112. Bapst, D.W. paleotree: An R package for paleontological and phylogenetic analyses of evolution. *Methods Ecol. Evol.* **2012**, *3*, 803–807. [CrossRef]
113. Bennett, D.J.; Sutton, M.D.; Turvey, S.T. treeman: An R package for efficient and intuitive manipulation of phylogenetic trees. *BMC Res. Notes* **2017**, *10*, 30. [CrossRef] [PubMed]
114. Burgstaller-Muehlbacher, S.; Crotty, S.M.; Schmidt, H.A.; Reden, F.; Drucks, T.; von Haeseler, A. ModelRevelator: Fast phylogenetic model estimation via deep learning. *Mol. Phylogenetics Evol.* **2023**, *188*, 107905. [CrossRef]
115. Sarkar, R. Low distortion delaunay embedding of trees in hyperbolic plane. In Proceedings of the International Symposium on Graph Drawing, Eindhoven, The Netherlands, 21–23 September 2011; pp. 355–366.
116. Matsumoto, H.; Mimori, T.; Fukunaga, T. Novel metric for hyperbolic phylogenetic tree embeddings. *Biol. Methods Protoc.* **2021**, *6*, bpab006. [CrossRef]
117. Jiang, Y.; Tabaghi, P.; Mirarab, S. Learning Hyperbolic Embedding for Phylogenetic Tree Placement and Updates. *Biology* **2022**, *11*, 1256. [CrossRef]
118. Macaulay, M.; Darling, A.; Fourment, M. Fidelity of hyperbolic space for Bayesian phylogenetic inference. *PLoS Comput. Biol.* **2023**, *19*, e1011084. [CrossRef]

119. Thirunavukarasu, A.J.; Ting, D.S.J.; Elangovan, K.; Gutierrez, L.; Tan, T.F.; Ting, D.S.W. Large language models in medicine. *Nat. Med.* **2023**, *29*, 1930–1940. [CrossRef] [PubMed]
120. Lubiana, T.; Lopes, R.; Medeiros, P.; Silva, J.C.; Goncalves, A.N.A.; Maracaja-Coutinho, V.; Nakaya, H.I. Ten quick tips for harnessing the power of ChatGPT in computational biology. *PLoS Comput. Biol.* **2023**, *19*, e1011319. [CrossRef] [PubMed]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.