Indo-European phylogenetics with R

A tutorial introduction

David Goldstein
University of California, Los Angeles, CA, USA
dgoldstein@humnet.ucla.edu

Abstract

The last twenty or so years have witnessed a dramatic increase in the use of computational methods for inferring linguistic phylogenies. Although the results of this research have been controversial, the methods themselves are an undeniable boon for historical and Indo-European linguistics, if for no other reason than that they allow the field to pursue questions that were previously intractable. After a review of the advantages and disadvantages of computational phylogenetic methods, I introduce the following methods of phylogenetic inference in R: maximum parsimony; distance-based methods (UPGMA and neighbor joining); and maximum likelihood estimation. I discuss the strengths and weaknesses of each of these methods and in addition explicate various measures associated with phylogenetic estimation, including homoplasy indices and bootstrapping. Phylogenetic inference is carried out on the Indo-European dataset compiled by Don Ringe and Ann Taylor, which includes phonological, morphological, and lexical characters.

Keywords

phylogenetics – computational methods – parsimony – UPGMA – neighbor joining – maximum likelihood – homoplasy – bootstrapping

1 Introduction

Phylogenetic trees model linguistic descent. More specifically, they are hypotheses about the order of lineage-splitting events from an often unobservable common ancestor to a set of observable descendants (Bowern & Koch 2004: 8–9, Pagel 2017: 152). The phylogeny of the Indo-European languages is a mat-

© DAVID GOLDSTEIN, 2020 | DOI:10.1163/22125892-20201000

ter of long-standing debate (for a recent overview, see Ringe 2017). Widmer (2018: 374) writes that "Auch in der Indogermanistik gibt es keinen Konsens, wie die Topologie des Stammbaums der indogermanischen Sprachen im Einzelnen aussieht."[1] The members of late clades are clear (that is, we are in no doubt about which languages belong to, e.g., the Celtic clade), but the order in which early clades formed has evaded consensus—with the notable exception of Anatolian, which is widely believed to be a sister to Proto-Nuclear-Indo-European:

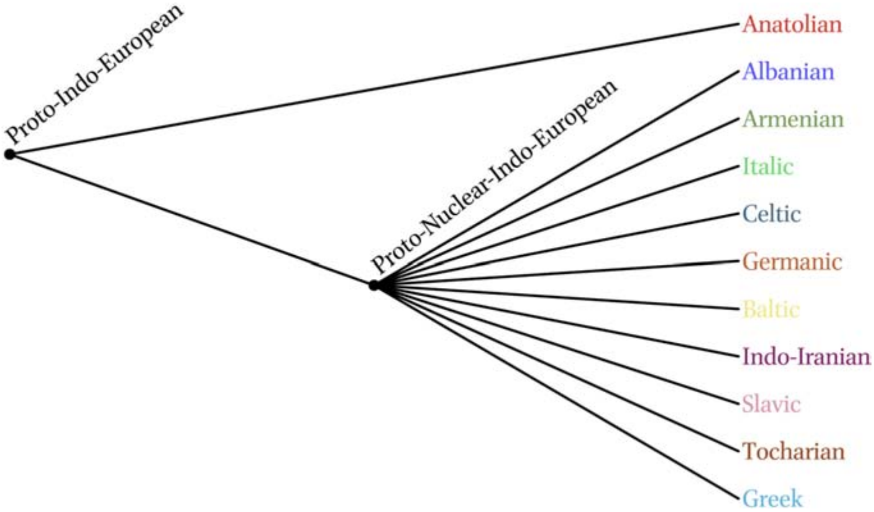


FIGURE 1 Nuclear IE star phylogeny

[1] It has of course long been debated whether tree graphs are the best way to represent linguistic history (Schmidt 1872, Schuchardt 1900, and more recently Enfield 2014: 1–8). I take up the question of the amount of phylogenetic structure in the Ringe-Taylor dataset in section 5.7 below. My focus on phylogenetic trees here should not be interpreted as support for the view that the history of the archaic Indo-European languages is best modeled with trees as opposed to networks (on which see, e.g., McMahon & McMahon 2005: 139–75, Nichols & Warnow 2008: 762–64, Huson, Rupp, & Scornavacca 2010, Morrison 2011, François 2015, Agee 2018). I do, however, subscribe to the view of Ringe (2017: 65) that "the tree model is always the better scientific hypothesis for any specific case, unless and until it becomes completely untenable." See also the recent defense of phylogenetic trees by Jacques & List (2018). There are of course cases where linguistic history is more adequately modeled with a network than a tree. Linguistic histories such as these can also be modeled computationally. For an example of a phylogenetic network analysis of the Indo-European languages, see Boc, Di Sciullo, & Makarenkov (2010). Schliep (2018a) provides an introduction to network analysis in `phangorn`.

One of the major questions of Indo-European linguistics is the order in which clades formed within Nuclear-Indo-European.

In the last twenty or so years, the methods of phylogenetic estimation have changed dramatically. Jäger (2015: 12752) goes so far as to declare: "Computational phylogenetics is in the process of revolutionizing historical linguistics."[2] In fact, the situation is more complex, and Jäger's statement premature.

On the one hand, it is true that computational phylogenetics has expanded the toolkit of historical linguistics. At the same time, the first wave of research in computational linguistic phylogenetics has engendered extensive controversy (see, e.g., Pereltsvaig & Lewis 2015 along with the reviews of Bowern 2017 and Verkerk 2017). It is no surprise then that skepticism towards computational linguistic phylogenetics runs high in certain circles (see, e.g., Heggarty 2006, Nichols & Warnow 2008: 760).

A more accurate assessment of the current status of computational phylogenetics is that it offers an enormous amount of potential. This potential does not necessarily lie in the ability to overturn long-standing conclusions of the field. Rather, these new methods enable Indo-Europeanists to investigate aspects of language change that were previously intractable (such as estimating branch lengths, rates of character change, and rates of diversification).

It is essential to understand both the advantages and disadvantages of the various computational phylogenetic methods (cf. Bowern 2018). Although it is possible to answer questions with computational methods that are otherwise intractable, computational methods are not in and of themselves "superior" to traditional methods. Reliable results can only come from the use of computational methods in concert with traditional analysis. Furthermore, although computational linguistic phylogenetics will undoubtedly yield exciting results, this success will not come at the expense of traditional comparative linguistic research, since the relationship between these two approaches is one of mutual symbiosis.

The goal of this article is to enable historical linguists who have no experience with computational methods to estimate phylogenies with R and RStudio (R Core Team 2019).[3] Although the focus of this tutorial is decidedly on

[2] Bowern (2018: 282) notes that the term *computational* (*linguistic*) *phylogenetics* is used in different senses in the literature. It is possible that Jäger in the quotation above is referring specifically to phylogenetic estimation with Bayesian-MCMC methods. I use *computational* (*linguistic*) *phylogenetics* to refer to methods of phylogenetic inference based on at least one of the following: an optimality criterion, an algorithm, or a stochastic trait model.

[3] For those new to programming or R, an introductory course may prove helpful. Such courses can be found online at no cost, such as that offered by Datacamp (https://www.datacamp.com/courses/free-introduction-to-r).

basic knowledge, I provide a substantial introduction to each method (maximum parsimony, UPGMA, NJ, and maximum likelihood).[4] The recent overview papers by Nichols & Warnow (2008), Dunn (2015), Bowern (2018), and Garrett (2018) make excellent companion pieces to the practical orientation of this article.

The remainder of the paper is organized as follows. Section 2 discusses the advantages and disadvantages of computational estimation of linguistic phylogenies. Section 3 introduces R and RStudio, the software that we will use for phylogenetic analysis. Building on this, section 4 introduces the dataset and guides the reader through the process of reading data into R. Sections 5 through 8 form the core of the article. These sections present parsimony methods, distance-based methods, and maximum likelihood methods of phylogenetic inference. Valedictory remarks bring the paper to a close in section 9.

Before discussing the advantages and disadvantages of computational linguistic phylogenetics, I need to say a word about the descriptive terms used throughout this paper. I generally prefer the terms current in evolutionary biology to those used in historical linguistics (a practice shared by Lass 1997). The former offers a much richer conceptual vocabulary for phylogenetic analysis than historical linguistics and I see no reason to pass on this bounty.[5] Following Ewens & Grant (2005: 497), I avoid the term (*phylogenetic*) *reconstruction* in favor of (*phylogenetic*) *estimation* or *inference*, since *reconstruction* suggests that the process of inferring past linguistic states is free of uncertainty, which is simply not the case. Claims about linguistic prehistory can rarely (if ever) be made with certainty. Concerning the phylogeny of the Indo-European languages, none of the trees in the literature (or presented below) is the true tree (cf. Garrett 2006: 43). The true tree is currently unknowable, because it is unclear how many branches or languages of Indo-European have vanished from the historical record. When it comes to phylogenies and ances-

[4] Bayesian-MCMC methods of phylogenetic inference are not covered in this tutorial. On account of the computational power these methods demand, R is not a practical option. A number of software options are available for Bayesian phylogenetics, including BEAST 2 (Drummond & Bouckaert 2015), RevBayes (Höhna et al. 2016), BEAST 2.5 (Bouckaert et al. 2019), or BEASTLing (Maurits et al. 2017).

[5] Introductions to phylogenetics include Wenzel 2002, Felsenstein 2004, Wiley & Lieberman 2011, Baum & Smith 2013, Hamilton 2013, and Hall 2018. The nature of the data from which Indo-Europeanists draw phylogenetic inferences is not unlike that of fossil data used in palaeontology, so it is particularly instructive to read the literature in palaeontology phylogenetics, e.g., Wiens 2000 and Mounce 2013. For more on the mathematics of phylogenetic analysis, see, e.g., Durbin et al. 1998, Semple & Steel 2003, Ewens & Grant 2005, Gascuel 2007, Sokal & Rohlf 1994, Yang 2014, and Steel 2016.

tral states, our goal is the best approximation of the true tree and the true state given the extant data.[6]

2 The advantages and disadvantages of computational methods

The dataset introduced below contains characters from 24 taxa (i.e., languages, or tips of the phylgenetic tree). The number of possible unrooted trees for this dataset is 563,862,029,680,583,512,791,449,600.[7] The number of possible rooted trees is 25,373,791,335,626,255,807,872,499,712 (Felsenstein 1978a, Felsenstein 2004: 19–36, Baum & Smith 2013: 187–90). In either case, the possible tree space is overwhelming. Although a specialist knows that wide swaths of this tree space are incorrect, it is nevertheless beyond human capabilities to assess which of the many viable candidate trees best fits the data.

It is well known that languages can emit weak or even conflicting phylogenetic signals. Phylogenetic algorithms enable us to make principled decisions on how to handle such cases. This is important, because in such cases researchers can be influenced by phylogenetic analyses that they want to be true. As Efron & Tibshirani (1993: 1) put it, “we are all too good at picking out non-existent patterns that happen to suit our purposes.” McMahon & McMahon (2005: 68–69) and Scarborough (2016: 33) discuss this issue in more detail.

Computational phylogenetics enables us to explore dimensions of linguistic history that are rarely if ever discussed in the traditional scholarship. The Indo-European literature has focused almost exclusively on the question of topology.[8] That is of course an important question, but there are other aspects of the history of the Indo-European languages that should also be pursued. For example, we know little about how the rates of change among different components of language (phonology, morphology, syntax, and the lexicon) vary over time (see Nettle 1999a, Nettle 1999b, Clackson 2000).

[6] Ancestral state inference (i.e., “linguistic reconstruction”) is possible in R, but lies beyond the scope of this tutorial. See for instance Paradis (2012: 247–58, 272, 276, 294, 297, 303) and `vignette("Ancestral")` in `phangorn`. For more on ancestral state inference in general, see Nunn (2011: 52–97), Yang (2014: 125–33) and Bowern (2018: 289–91).

[7] An unrooted tree is a phylogenetic tree without a defined root. Unrooted trees provide no information about the temporal sequence of lineage-splitting events. See further Baum & Smith (2013: 61–64).

[8] The topology of a phylogenetic tree is the relative order of its branches. Tree topology typically tells us how closely related two languages are. See further Baum & Smith (2013: 45–47).

Computational methods also enable researchers to assess the extent to which the data provide evidence for a particular clade. This is absolutely crucial to any phylogenetic analysis. In making inferences about events that reach back several millennia in time, we do not deal in certainties. We therefore need tools that enable us to acknowledge this uncertainty and the limitations of our data:

> The field of phylogenetics should not be seen as an attempt to *build* trees, but rather to examine alternative trees and then quantify the extent to which data support or reject different phylogenetic conclusions.
>
> BAUM & SMITH (2013: 265), emphasis in original

To this end, I introduce bootstrap analysis in section 6 below.

Finally, computational methods—in particular maximum likelihood estimation and Bayesian inference—enable historical linguists to infer phylogenies based on specific models of linguistic change (known as TRANSITION MODELS; see section 8.3 below). Such models encode assumptions, for instance, about the probability of change and whether certain directions of change are more or less likely. With these methods, it thus becomes possible to incorporate a theory of language change into phylogenetic inference.

For all the advantages of computational methods, they are not without their pitfalls, perhaps the most threatening of which is the tendency to confuse model sophistication (or model precision) with model accuracy (cf. Pereltsvaig & Lewis 2015: 7–10 on SCIENTISM). Simply because the sophistication of computational phylogenetic methods outstrips that of traditional methods, one might come to think that these methods (in particular Bayesian inference) will automatically yield a superior approximation to the true tree. Another concern along similar lines is that computational methods can lead to researcher absenteeism in as much as it can lead one to think that computational power can make up for datasets that are either flawed or characterized by conflicting phylogenetic signals. That is of course impossible. The computational methods presented below are only as good as the data culled for analysis.

Some have argued that the transmission of genes is fundamentally different from the transmission of linguistic knowledge (e.g., Andersen 2006, Lewis & Pereltsvaig 2012, Pereltsvaig & Lewis 2015: 149–56).[9] Armed with such a view,

[9] It is, however, easy to find both biologists and linguists who emphasize the similarities of biological evolution and linguistic change, e.g., Darwin (1882: 90), Atkinson & Gray (2005), Croft (2008), Pagel (2009), Borchsenius, Daval-Markussen, & Bakker (2017), Pagel (2017). Despite these similarities, it remains unclear whether we should adopt an "evolutionary"

one might question whether the computational methods that have been developed for the phylogenetic estimation of species are suitable for linguistic data (see Bowern 2018: 283–84). What unites evolutionary biology and historical linguistics is not so much the phenomena that they investigate, but rather the nature of the questions that they pursue. Both fields aim to draw inferences about prehistory from observable data. Provided that the models and underlying assumptions are compatible with linguistic change, there is no reason why methods developed for the evolution of species should be unsuitable for linguistic history. Pagel (2017: 152) draws attention to the crucial point that both genetic information and linguistic properties can be represented as digital systems of inheritance (cf. Bowern 2018: 284). It is true that some methods or models developed for evolutionary biology will not be applicable to linguistic data, but one cannot conclude from such incompatibility that methods of computational phylogenetics in general cannot be used on linguistic data.

2.1 *Computational phylogenetics and traditional subgrouping*

If one accepts the need for computational phylogenetics, the question arises of what the relationship between computational and traditional methods should be. Computational linguistic phylogenetics faces the following conundrum. If the methods produce novel results at odds with traditional subgrouping, they may be dismissed as incorrect (the most salient example of this is the debate that has surrounded Gray & Atkinson 2003 and Bouckaert et al. 2012). If the methods recapitulate the results of traditional analyses, then they may be deemed otiose. Consequently, one can come away with the impression that there is no place in the field for computational methods, in as much as they are at best unnecessary and at worst misguided.

First and foremost, computational methods should not be viewed as a replacement of traditional subgrouping as based on the comparative method (Ringe, Warnow, & Taylor 2002: 66, Bowern 2017: 427). Computational methods should be used in conjunction with the traditional methods known to yield reliable results:

> [T]raditional subgrouping is logically coherent and methodologically unobjectionable: in order to subgroup a particular subset of the family's languages together, one demands that they exclusively share clear and linguistically significant innovations which are unusual enough that they

approach to language of the sort advocated by, e.g., Schleicher (1863), Lass (1997), Croft (2008), Rosenbach (2008), or Pagel (2017). See further the papers in Hoenigswald & Weiner (1987) and Eckardt, Jäger, & Veenstra (2008).

> could not reasonably have arisen more than once independently. To put it in biologist's terms, one recognises a clade by the presence of unique synapomorphies, rigorously excluding any traits that might conceivably be analogous rather than homologous. This is so clearly correct that we have no intention of even questioning it.[10]
>
> RINGE, WARNOW, & TAYLOR (2002: 65–66)

There are various ways in which traditional subgrouping and computational phylogenetics can complement one another. For instance, computational methods can play a confirmatory role. If computational methods come to the same answers that the field achieved without the aid of a computer, that is worth knowing. (It would be worth knowing because it would mean that we have an algorithm that approximates the method of phylogenetic inference among historical linguists.) In a similar vein, if some of the phylogenetic analyses are at odds with computational results, that is also important. In addition, computational methods can be used to guide us out of an impasse. There are many aspects of the history of the archaic Indo-European languages for which traditional methods have not yet yielded a consensus answer. As the quotation from Widmer above reveals, there is a lot of uncertainty surrounding the topology of Indo-European, for instance.

Subgroups are standardly established on the basis of shared innovations. To identify an innovation one has to be able to identify an ancestral state. In some cases, this is not a challenge. For instance, given a language with only oral vowels and nasal consonant codas and a related language with nasal vowels but no nasal consonant codas, the nasal vowels of the latter are very likely the innovation. In other cases, determining the innovation is more challenging. The continued uncertainty of whether the augment was present in PIE is one such example.[11]

Not only does subgrouping depend on the inference of ancestral states, but the inference of ancestral states also depends upon subgrouping. When a cognate lexical item is attested in, say, three taxa then one has to decide how far back its ancestral lexical item should be projected—that is, whether to some intermediate interior node or to PIE itself. Phylogeny plays a crucial role in assessing such questions (for further discussion, see, e.g., Mallory &

[10] It is worth noting that Babel et al. (2013) challenge this allegedly unassailable principle (see further Lass 1997: 143–59).

[11] The augment is a morpheme prefixed to certain finite verbal forms. For its distribution in archaic Indo-European, see example (1) in section 4 below.

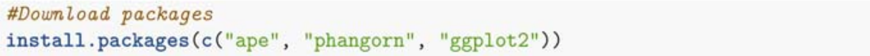
Adams 2006: 106–10, Olander 2018). The upshot is a chicken-and-egg scenario in which subgrouping and ancestral-state inference can be mutually dependent endeavors.

3 Software

R is a statistical programming language built on the S language (Wickham 2014). R offers many advantages, foremost of which is that it is free, general purpose software. It boasts over 4,000 libraries, which include a wide array of packages for phylogenetic analysis. The analyses and tree graphs presented below were all carried out in R version 3.5.3.[12] R can be downloaded at https://www.r-project.org.

Once R has been installed, one should also download the Integrated Development Environment (IDE) RStudio, which is available at https://www.rstudio.com.[13] I urge the reader to use RStudio (as opposed to R) for carrying out the phylogenetic analyses below.

Once you have R and RStudio installed, you will need to install packages for phylogenetic analysis. The two most important packages for our purposes are ape (Paradis 2012) and phangorn (Schliep 2011, Schliep 2018b). Packages can be downloaded to your hard drive with the following command (the '#' symbol is used for comments in R; entering them in the R console in RStudio will have no effect):



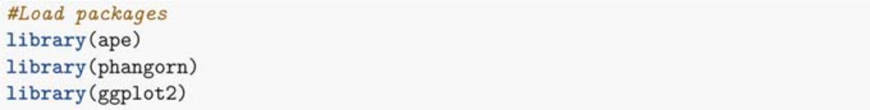
```
#Download packages
install.packages(c("ape", "phangorn", "ggplot2"))
```

Typically you will download packages from CRAN, The Comprehensive R Archive Network (https://cran.rstudio.com). As explained below, however, packages can be downloaded from other sources, such as BioConductor or GitHub.

Once the packages have been downloaded, they need to be loaded into the current session, which can be done with the `library()` function:

[12] It is worth noting that R is not the only software with which one can infer phylogenies. Among programming languages, there is also Python. Johann-Mattis List has in fact developed a range of Python software for historical linguistics including phylogenetic analysis (e.g., List 2017). His website contains a wealth of information and resources: http://lingulist.de. Egan (2006: 81) and Felsenstein (n.d.) also list other software for phylogenetic analysis.

[13] For introductions to using R for linguistic analysis, see Gries (2013), Levshina (2015), and Gries (2017). Grolemund & Wickham (2017) is an introduction to R based on a suite of packages known as the Tidyverse (https://www.tidyverse.org).

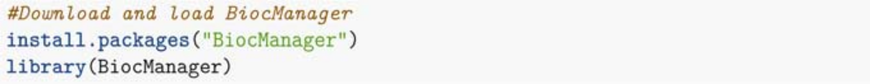


```
#Load packages
library(ape)
library(phangorn)
library(ggplot2)
```

Once the packages are loaded into your working environment, their functions will be at your disposal.

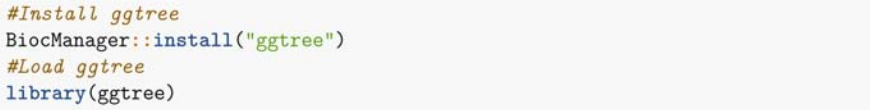
At this point, you may want to create a new script file in RStudio rather than work directly in the R console. To do this, open RStudio and go to `File` > `New File` > `R Script` in the menu bar. A new script file will then appear above the console pane. You should put the commands for loading the above packages in the preamble of the document. All of the code below for phylogenetic inference and visualization of trees is available along with the datasets used in this tutorial at http://doi.org/10.5281/zenodo.3417299.

For plotting trees, one can use the packages `ggdendro` and `ggtree` (Yu et al. 2017), which extend the `ggplot2` package. The trees below were produced with version `ggtree` version 1.14.6 (Yu et al. 2017). In contrast to the other packages described in this tutorial, `ggtree` is not available on CRAN. It is available from BioConductor, which can be downloaded with the following code:



```
#Download and load BiocManager
install.packages("BiocManager")
library(BiocManager)
```

Once `BiocManager` is loaded, `ggtree` is installed and loaded as follows:



```
#Install ggtree
BiocManager::install("ggtree")
#Load ggtree
library(ggtree)
```

4 The dataset

The phylogenetic trees presented in the subsequent sections are based on the phonological (Ringe & Taylor 2007b), morphological (Ringe & Taylor 2007a), and lexical characters (Ringe & Taylor 2002, Ringe, Warnow, & Taylor 2012) in the screened dataset created by Don Ringe and Ann Taylor (Nakhleh, Ringe, & Warnow 2005: 178; for a critical assessment of the dataset, see Drinka 2013: 383–85). It contains twenty-two phonological characters; twelve morphological ones; and 259 lexical characters, for a total of 293 characters.

The dataset uses multistate character values. The augment, which is character M2 in Ringe & Taylor (2007a), will serve as an illustrative example (for further examples, see Nakhleh, Ringe, & Warnow 2005: 410–18):

(1) Multi-state character encoding for the augment

Hittite	2	Avestan	1	Luvian	10	Gothic	15
Armenian	1	Old Church Slavic	5	Lycian	11	Old Norse	16
Greek	1	Lithuanian	6	Tocharian A	12	Old High German	17
Albanian	3	Old English	7	Old Persian	1	Welsh	18
Tocharian B	4	Old Irish	8	Old Prussian	13	Oscan	19
Vedic	1	Latin	9	Latvian	14	Umbrian	20

The value 1 denotes the presence of the augment. Character values from 2 onwards denote its absence.[14]

At the risk of stating a truism, I want to stress the critical importance of character selection and encoding (cf. Nakhleh et al. 2005: 172, Geisler & List 2010).[15] This is by far the most important component of phylogenetic analysis. No matter the sophistication of the method of phylogenetic inference, if the linguistic analysis of the data is flawed (e.g., incorrect coding of cognates or poorly selected characters), the estimated phylogeny will also be flawed (cf. Johnson 2008: 250, Chang et al. 2015: 221). In an era of ever increasing technological sophistication, it is more important than ever that we be able to distinguish accuracy and precision, two phenomena that, though often mistaken for one another, are in fact worlds apart.[16] Simply because a method is more sophisticated or yields more precise answers (e.g., an estimated time depth for Proto-Indo-European) does not mean that such answers automatically lay greater claim to the truth.

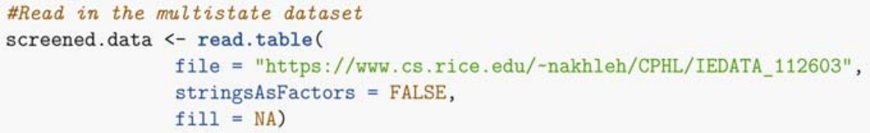
[14] According to Ringe & Taylor (2007a: 3), the absence of the augment is not coded with a unique value because that would imply a historically shared change.

[15] For discussion of various aspects of characters and character selection, see Taylor, Warnow, & Ringe (2000), Kessler (2001), Ringe, Warnow, & Taylor (2002: 71–73), Wichmann & Saunders (2007), Nichols & Warnow (2008: 764–66), Chang et al. (2015: 200–04), Pereltsvaig & Lewis (2015: 218–28), Scarborough (2016: 186–88), Bowern (2018: 287–88).

[16] In science, engineering, and statistics, there are technical definitions of accuracy and precision. Accuracy is generally defined as how close the measurement of a quantity is to its true value. Precision, on the other hand, refers to variability in measurement.

4.1 *Reading data into R*

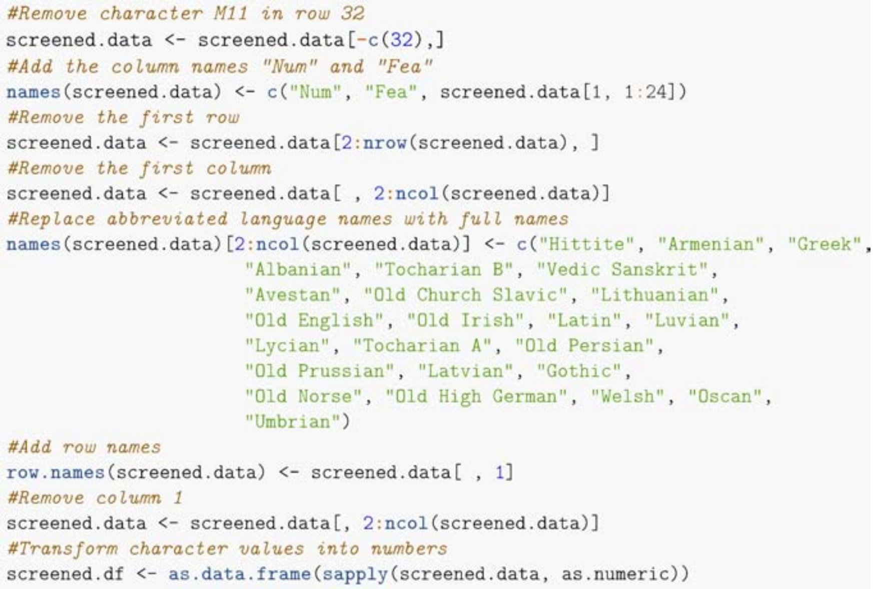
The Indo-European character datasets curated by Don Ringe and Ann Taylor are available on Luay Nakleh's website at https://www.cs.rice.edu/~nakhleh/CPHL/.[17] We read the data into R from the web as follows:



```
#Read in the multistate dataset
screened.data <- read.table(
                file = "https://www.cs.rice.edu/~nakhleh/CPHL/IEDATA_112603",
                stringsAsFactors = FALSE,
                fill = NA)
```

The Indo-European character data is now the R object `screened.df` (the object bears the extension `.df` because it is a data structure known as a dataframe). The argument `stringsAsFactors = FALSE` enables the values in the table to be treated as character strings and `fill = NA` is needed because the rows do not all have the same number of elements. This argument inserts `NA` in cells of the table to make the rows equal in length.

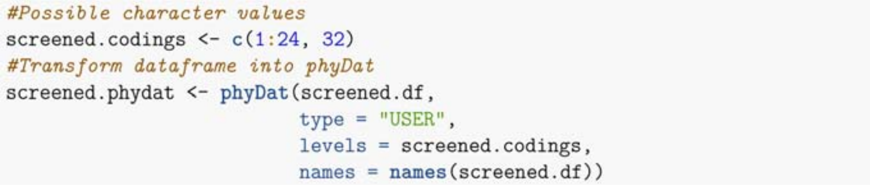
A few things need to be changed before we can analyze the data (character M11 is removed per Ringe & Taylor 2007a: 9–10):



```
#Remove character M11 in row 32
screened.data <- screened.data[-c(32),]
#Add the column names "Num" and "Fea"
names(screened.data) <- c("Num", "Fea", screened.data[1, 1:24])
#Remove the first row
screened.data <- screened.data[2:nrow(screened.data), ]
#Remove the first column
screened.data <- screened.data[ , 2:ncol(screened.data)]
#Replace abbreviated language names with full names
names(screened.data)[2:ncol(screened.data)] <- c("Hittite", "Armenian", "Greek",
                       "Albanian", "Tocharian B", "Vedic Sanskrit",
                       "Avestan", "Old Church Slavic", "Lithuanian",
                       "Old English", "Old Irish", "Latin", "Luvian",
                       "Lycian", "Tocharian A", "Old Persian",
                       "Old Prussian", "Latvian", "Gothic",
                       "Old Norse", "Old High German", "Welsh", "Oscan",
                       "Umbrian")
#Add row names
row.names(screened.data) <- screened.data[ , 1]
#Remove column 1
screened.data <- screened.data[, 2:ncol(screened.data)]
#Transform character values into numbers
screened.df <- as.data.frame(sapply(screened.data, as.numeric))
```

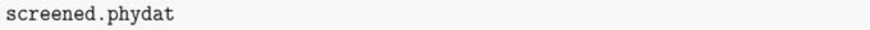
[17] Other IE datasets are available from IELex (http://ielex.mpi.nl) and in the supplementary files of Chang et al. (2015) (https://muse.jhu.edu/article/576999).

For several of the phylogenetic analyses below, I use version 2.4 of the package phangorn (Schliep 2018b), which requires that the data be in the phyDat structure. The following code transforms the above dataframe into a phyDat object (see further Schliep 2017):



```
#Possible character values
screened.codings <- c(1:24, 32)
#Transform dataframe into phyDat
screened.phydat <- phyDat(screened.df,
                          type = "USER",
                          levels = screened.codings,
                          names = names(screened.df))
```

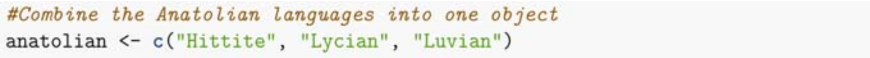
The object screened.phydat will serve as the input to most of the phylogenetic analyses below. To see what the object contains just type its name into the console:



```
screened.phydat
```

```
## 24 sequences with 293 character and 282 different site patterns.
## The states are 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 32
```

Finally, most of the methods below infer unrooted trees. To establish the branching order among clades, we need to select an outgroup. Since Anatolian is now agreed by many to have been the first clade to branch off (e.g., Melchert & Oettinger 2009: 53–54, Melchert forthcoming), the Anatolian languages in the dataset (that is, Hittite, Lycian, and Luvian) will serve as the outgroup. It is created as follows:



```
#Combine the Anatolian languages into one object
anatolian <- c("Hittite", "Lycian", "Luvian")
```

Below I use the object anatolian in the specification of the outgroup.

5 Parsimony methods

We begin with parsimony methods (Fitch 1971, Stewart 1993, Swofford et al. 1996: 415–26, Kitching et al. 1998, Felsenstein 2004: 1–146, Albert 2005, Swofford & Sullivan 2009, Nunn 2011: 30–33, Baum & Smith 2013: 173–215, Yang 2014: 95–100, Warnow 2018: 63–69), which resemble traditional methods of subgrouping. Maximum parsimony methods are based on an optimality criterion: the tree that requires fewest changes for a given dataset is optimal. (The total number of steps for a dataset on a given tree is known as the LENGTH of the tree.) More specifically, the optimal tree minimizes the amount of homoplasy.[18] Underlying this method is the assumption that language change is slow (in the sense that the characters have only undergone a small number of transitions) and that we should therefore prefer phylogenies that minimize the number of changes posited for the data.[19]

There are several algorithms for calculating the parsimony score of a tree for a given dataset, the most prominent of which are Fitch, Sankoff, and Dollo. In Fitch parsimony, a change between any two states is possible, and all changes count for just one step (Fitch 1971, Felsenstein 2004: 11–13). Sankoff parsimony also allows a change between any two states (Sankoff 1975, Felsenstein 2004: 13–16). The crucial difference is that Sankoff parsimony assumes a cost matrix for transititions between any two given states.[20]

Another form of parsimony that is relevant to linguistic phylogenetics is Dollo parsimony. According to this model, a trait can be acquired once, and if lost it can never be regained (Farris 1977). This form of parsimony is of interest to historical linguistics because it has a correlate in the domain of sound change, namely GARDE'S PRINCIPLE (Garde 1961), which states that phonological mergers cannot be undone (Hoenigswald 1960: 75–82, 87–98). So once two phonemes merge, their ancestral distribution cannot be recovered. (For a discussion of this phenomenon and apparent exceptions, see Silverman 2012: 62–77.)

For up to about twenty taxa, the branch and bound algorithm (introduced in section 5.1 below) is guaranteed to find the most parismonious tree. For larger datasets, we need recourse to a HEURISTIC SEARCH algorithm, which I introduce in section 5.5 below. In contrast to the branch and bound methods, these search algorithms are not guaranteed to find the most parsimonious tree.

[18] HOMOPLASY refers to situations in which a character state arises more than once on a tree. This includes both parallel independent innovations and character state-reversals (otherwise known as BACKMUTATION). See further Baum & Smith (2013: 93–95) and section 5.6 below.

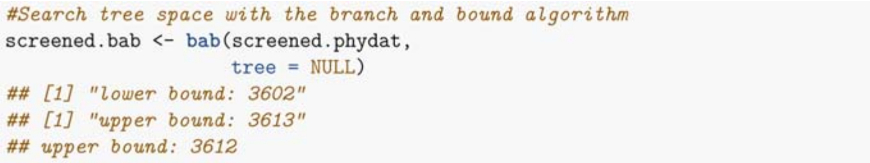
[19] For maximum parsimony analyses of archaic Indo-European languages, see Rexová, Frynta, & Zrzavý (2003), Skelton (2015), and DeLisi (2018). Outside of Indo-European, see Holden (2002) on Bantu and Baxter (2006) on Chinese.

[20] Such a cost matrix enabled biologists to assign different weights to transitions (a change between two purines or two pyrimidines) and transversions (a change between a purine and a pyrimidine or between a pyrimidine and a purine). The idea of assigning weights to linguistic changes is appealing (Bowern & Koch 2004: 4, Nakhleh et al. 2005: 180, 188–89), but it is unclear how values for a cost matrix should be assigned. For an example of analyses using weighted characters, see Nakhleh et al. (2005) and Skelton (2015).

5.1 *Branch and bound*

The branch and bound algorithm is guaranteed to find the most parsimonious tree(s) (Felsenstein 2004: 38). The algorithm does not, however, calculate the length of all possible trees, but rather exploits the following insight to exclude regions of unparsimonious trees (see further Felsenstein 2004: 60–64): adding taxa to a tree will never decrease its length (Baum & Smith 2013: 189, Huson, Rupp, & Scornavacca 2010: 35). That is, whatever homoplasy exists on a tree will never be reduced by adding taxa to the tree. So if removing taxa from a tree results in a parsimony score higher than that of the current bound (i.e., the current best tree), then all trees derived from this reduced tree will be less parsimonious (Baum & Smith 2013: 189). Thus the branch and bound algorithm reduces the tree space by eliminating swaths that cannot contain an optimal tree and thereby drastically reduces the number of trees for which a parsimony score is calculated.

The main disadvantage of this technique is that it is very slow and can only really be used for datasets that contain at most ten to twenty taxa. The package `phangorn` contains the function `bab()`, which will find all most parsimonious trees from a given dataset (depending on your computer, you may have to wait up to ten minutes to get the command prompt back):



```
#Search tree space with the branch and bound algorithm
screened.bab <- bab(screened.phydat,
                    tree = NULL)
## [1] "lower bound: 3602"
## [1] "upper bound: 3613"
## upper bound: 3612
```

With the `bab()` function, one can specify a start tree (i.e., a tree used to initiate the search) by adding `tree =` inside the parentheses. (Options of a function such as this one are known as ARGUMENTS.) Here I opted not to do that by setting the value of this argument to `NULL`. Doing so causes a ratchet search (introduced below in section 5.5) to be performed to find a start tree.

The output of the `bab()` function is an object of the class `multiPhylo` (see further Paradis 2012: 55–56). For our dataset, the branch and bound search returns fifteen maximally parsimonious trees. By calling the function `parsimony()` from the `phangorn` package (see further Paradis 2012: 165–66), we can confirm that the parsimony scores (or p-scores) are identical:[21]

[21] With the argument `method`, one can specify `"fitch"` or `"sankoff"` parsimony when calling the `parsimony()` function. Dollo parsimony can be calculated with the function `Rdollop()` in the package `RPHylip` (Revell & Chamberlain 2015).

```
#Calculate parsimony score for branch and bound trees
parsimony(screened.bab, screened.phydat)
##  [1] 3612 3612 3612 3612 3612 3612 3612 3612 3612 3612 3612 3612 3612 3612
## [15] 3612
```

There are fifteen p-scores, one for each tree.

5.1.1 Rooting the trees and adding branch lengths

The branch and bound algoritm returns unrooted trees, which we can confirm with the function `is.rooted()`:

```
#Check if the trees are rooted
is.rooted(screened.bab)
##  [1] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## [12] FALSE FALSE FALSE FALSE
```

To root the trees we call the function `root()`:

```
#Root the tree with Anatolian as the outgroup
screened.bab.rooted <- root(screened.bab,
                            outgroup = anatolian,
                            resolve.root = TRUE)
```

This code sets Anatolian as the outgroup of each of the trees from the branch and bound algorithm. To check that the trees are in fact rooted, we again call the function `is.rooted()`:

```
#Check if the trees are rooted
is.rooted(screened.bab.rooted)
##  [1] TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE
## [15] TRUE
```

The trees produced by the branch and bound algorithm also lack branch lengths. To add branch lengths to the trees, we call `acctran()` from the `phangorn` package:

```
#Add branch lengths
screened.bab.rooted.blength <- acctran(screened.bab.rooted,
                                       screened.phydat)
```

This function estimates branch lengths via a method known as ACCELERATED TRANSFORMATION. Homoplastic characters can lead to multiple maximally parsimonious trees. The central idea of accelerated transformation is to assign character-state changes as soon as possible on the tree, which maximizes character-state reversals (for more on the calculation of branch length, see Swofford & Maddison 1987, Felsenstein 2004: 70–72).

The following code returns the length of each branch on the first tree:

```
screened.bab.rooted.blength[[1]]$edge.length
##   [1] 136.0 103.0 115.5  94.5  61.0  54.0 107.0  73.0 155.5  48.5  72.5
## [12]  40.5  22.5  16.5  22.0  34.0  61.0  31.0  79.5  79.5 117.0 111.0
## [23] 151.5  48.5  72.0 118.0  54.5 111.5  59.0  41.0  69.5  94.5  68.5
## [34]  67.5  99.0  58.0  45.5 169.5  45.5  78.5  48.5  71.5  81.5 114.5
## [45] 104.5 104.5
```

The branch lengths represent the number of inferred changes. By changing the index in the double brackets, one can obtain the branch lengths for other trees. Summing the length of each branch, we obtain the p-score observed above:

```
sum(screened.bab.rooted.blength[[1]]$edge.length)
## [1] 3612
```

5.2 *Visualization*

Phylogenetic trees can be plotted with the `plot()` function. Here for instance is the first of the branch and bound trees:

```
#Plot the first branch-and-bound tree
plot(screened.bab.rooted.blength[[1]])
title("Branch and bound tree 1")
```

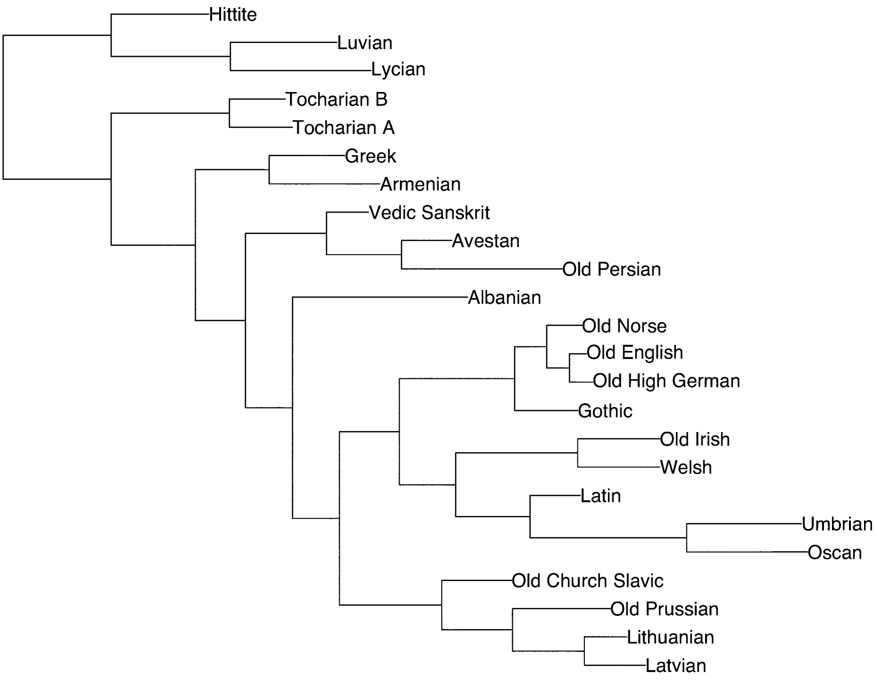


FIGURE 2 Branch and bound tree 1

The output will appear in the plot pane in the lower right corner of the RStudio console. By clicking the `Export` tab, one can save it as a file.[22]

Trees two and six of the the branch and bound trees are plotted below. To the right of each tree I include a heatmap of the phonological and morphological characters in the dataset so that one can get a sense of the underlying data. Tree two is paired with the phonological characters from the dataset, while tree six is paired with the morphological. In the interest of enhancing the visualization, the original multistate characters were transformed into binary characters. The binary dataset and the code used for the transformation are available at http://doi.org/10.5281/zenodo.3417299.

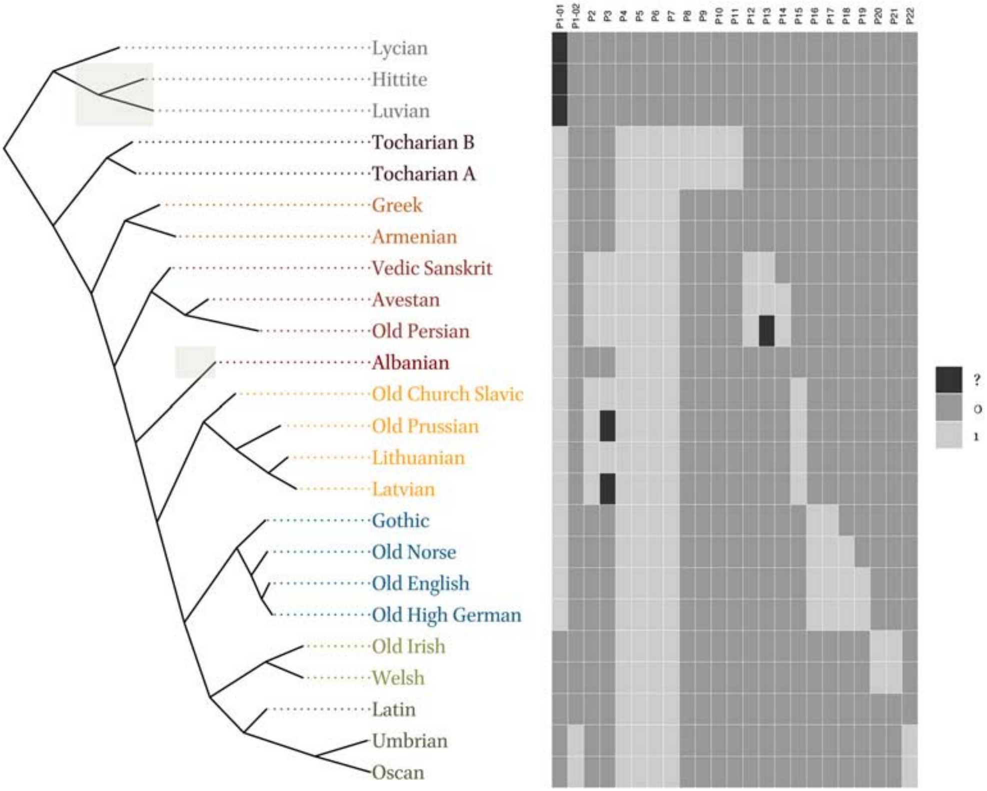


FIGURE 3 Branch and bound tree 2 with binary phonological characters

[22] If one wants to plot a tree in a different program, such as FigTree (http://tree.bio.ed.ac.uk/software/figtree/), one can export it with the `write.tree()` function. For the arguments of this function, type `?write.tree` into the console. The tree will be saved in Newick format in the working directory. (The Newick format is a standard way of representing trees.)

In the first three rows of the heatmap, the Anatolian languages show an almost uniform block of 0 values. We see in characters P4 through P7 some of the innovations (i.e., 1 values) that define Proto-Nuclear-Indo-European. (For a description of the change represented by each column, see Ringe & Taylor 2007b and Ringe & Taylor 2007a.)

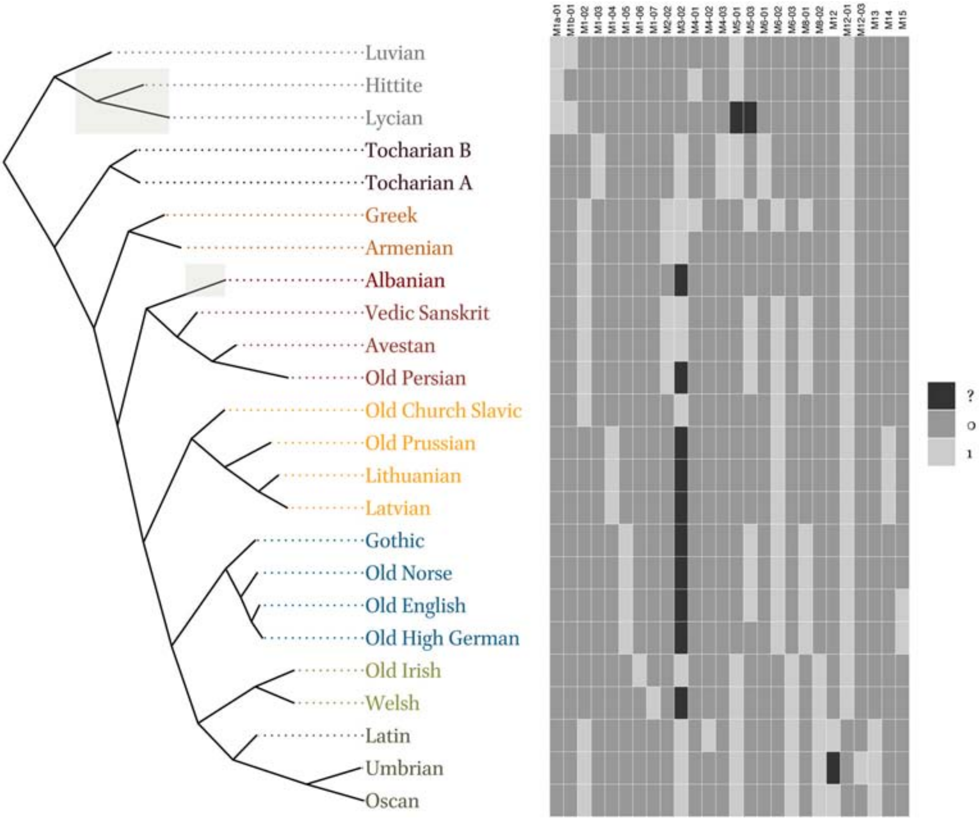


FIGURE 4 Branch and bound tree 6 with binary morphological characters

I have highlighted a portion of the Anatolian clade and Albanian because these are the loci of variation among the fifteen branch and bound trees. In the next section, I explore these fifteen branch and bound trees further with consensus and maximum clade credibility trees.

Given that the true Indo-European tree is not known, evaluation of phylogenetic methods is challenging (Nichols & Warnow 2008: 760). Since there is no debate among Indo-Europeanists about the members of clades such as Slavic, Celtic, and Germanic, below I use correct assignment of languages to recognized clades as the baseline evaluation measure.

5.3 *Maximum clade credibility tree*

We can summarize the set of branch and bound trees with a MAXIMUM CLADE CREDIBILITY TREE. The function `maxCladeCred()` evaluates each tree according to the frequency of each clade within the set of trees.

```
#Calculate maximum clade credibility tree
screened.bab.rooted.blength.mcc <- maxCladeCred(screened.bab.rooted.blength)
```

Trees with clades that are more frequent will have higher scores. The tree with the highest score is then selected as the maximum clade credibility tree:

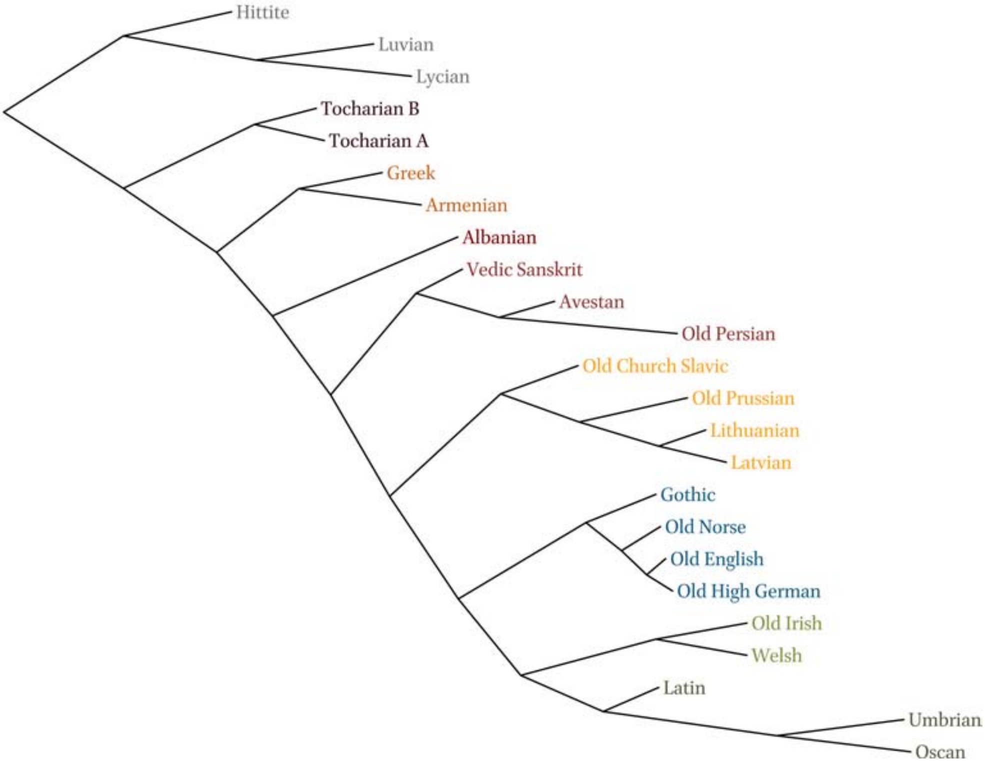


FIGURE 5 Maximum clade credibility tree from branch and bound search

5.4 *Consensus trees*

Another way to summarize a set of trees is with a CONSENSUS TREE (Paradis 2012: 179–82), which reduces a set of trees to a single tree. There are two types of consensus trees, strict consensus trees and majority-rule consensus trees. In a strict consensus tree, the clades that are not observed in all the trees of a set are represented as POLYTOMIES, that is, as multifurcating branches.[23] In a majority-rule consensus tree, the clades not observed in a majority of trees are represented as polytomous. To create a consensus tree, use the ape function consensus(). By default, a strict consensus tree is calculated:

```
#Calculate strict consensus tree
screened.bab.rooted.blength.strict <-
ape::consensus(screened.bab.rooted.blength)
```

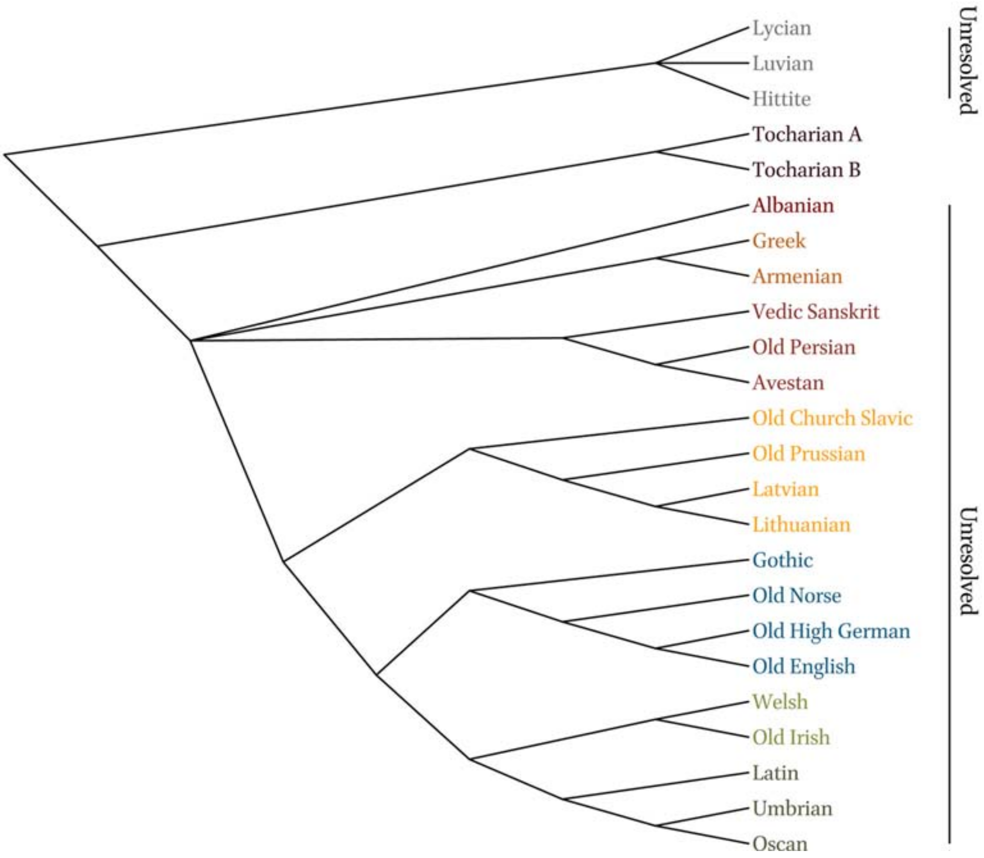
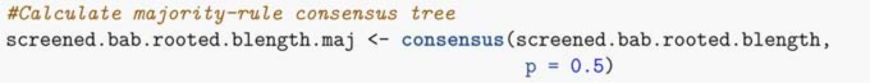


FIGURE 6 Strict consensus tree from branch and bound search

[23] A polytomy is a node with more than two descendant branches. It is otherwise known as a MULTIFURCATION. A HARD POLYTOMY is a lineage that splits into multiple descendants around the same time. A SOFT POLYTOMY reflects uncertainty about the true topology. That is, the multifurcation is not necessarily an accurate representation of the past. Most Indo-Europeanists would presumably characterize the consensus trees above as soft polytomies, since it is unlikely that the non-Anatolian archaic Indo-European languages all split up more or less simultaneously.

The length of each branch is now uniform because this particular tree was not among the branch and bound trees. (In fact, attempting to calculate the branch lengths of this tree with `acctran()` will yield an error message that the tree must be binary.)[24] The multifurcations reveal uncertainty at a number of points in the tree, in particular with the internal structure of Anatolian and the order of lineage-splitting events among Albanian, Greco-Armenian, Indo-Iranian, and the clade comprising Balto-Slavic, Germanic, and Italo-Celtic.

To calculate a majority-rule consensus tree, use the argument `p = 0.5`:



```
#Calculate majority-rule consensus tree
screened.bab.rooted.blength.maj <- consensus(screened.bab.rooted.blength,
                                             p = 0.5)
```

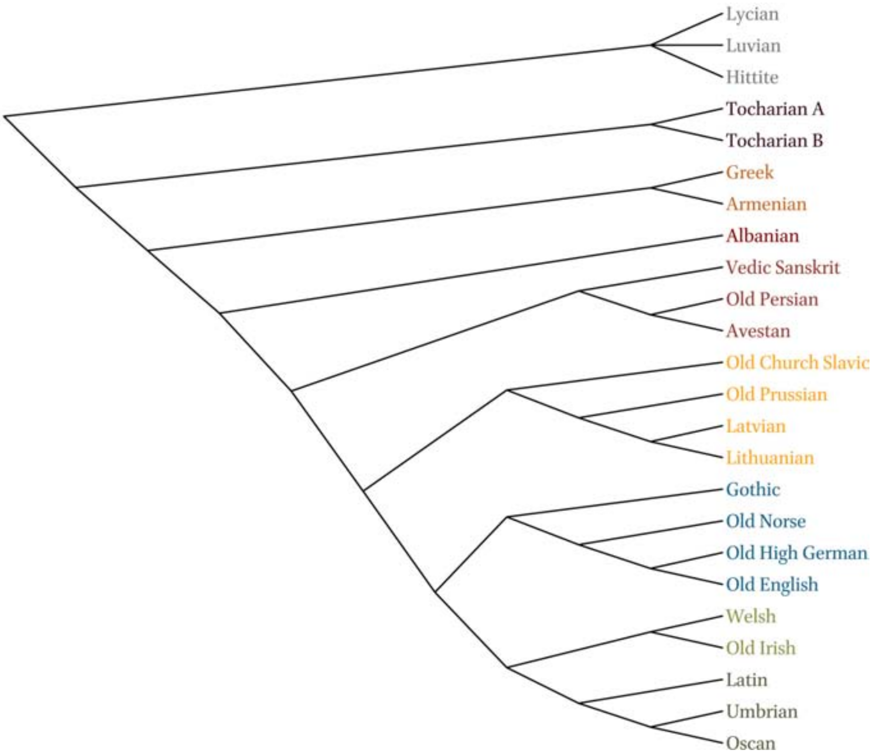


FIGURE 7 Majority-rule consensus tree from branch and bound search

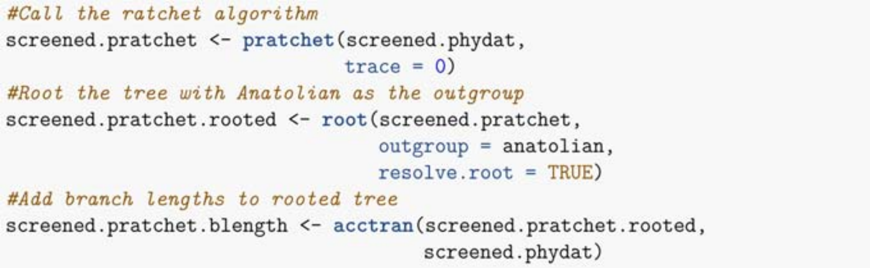
[24] There are ways to add branch lengths to strict consensus trees, but these will not be covered here. See https://github.com/bomeara/utilitree/.

This tree contains all of the clades that occur in at least fifty percent of the branch and bound trees. The branches are all now bifurcating with the exception of Anatolian.

5.5 *Heuristic search*

With large datasets, the size of the possible tree space makes it unfeasible to calculate the p-score of each tree. Various heuristic searches have therefore been developed. In phangorn these rely on branch-swapping methods. The basic idea behind such methods is to generate a number of trees by rearranging parts of an original tree and then moving to the one that has the best parsimony score (see further Huson, Rupp, & Scornavacca 2010: 37–40). This process is iterated until no improvement in the length of the tree can be found. The reader should be aware that the heuristic searches below are not guaranteed to find the most parsimonious tree(s), since there is the possibility that they can get stuck in local optima (roughly speaking, local optima are regions of the tree space that are good relative to other areas, but not the best).[25]

The phangorn package implements a parsimony-based heuristic search known as the ratchet. The ratchet search relies on a branch-swapping algorithm known as TREE BISECTION AND RECONNECTION (TBR). I refer the reader to Nixon (1999) and Felsenstein (2004: 51–52) for the details of the algorithm. The following code estimates a maximum parsimony tree with a ratchet search (which returns unrooted trees):



```
#Call the ratchet algorithm
screened.pratchet <- pratchet(screened.phydat,
                              trace = 0)
#Root the tree with Anatolian as the outgroup
screened.pratchet.rooted <- root(screened.pratchet,
                                 outgroup = anatolian,
                                 resolve.root = TRUE)
#Add branch lengths to rooted tree
screened.pratchet.blength <- acctran(screened.pratchet.rooted,
                                     screened.phydat)
```

[25] One therefore needs to replicate the results of a heuristic search using random starting points.

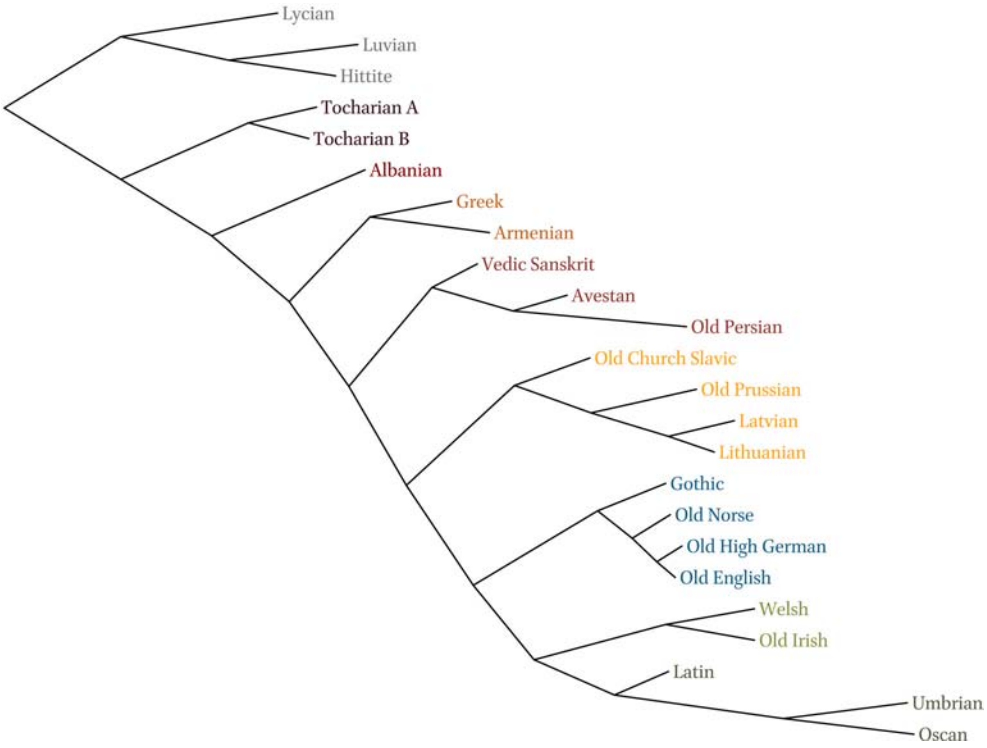


FIGURE 8 Parsimony ratchet tree

The parsimony ratchet is generally considered the most reliable among the branch-swapping heuristic search methods.

Two other branch-swapping algorithms are implemented in phangorn: NEAREST NEIGHBOR INTERCHANGES (NNI; Felsenstein 2004: 38–41, Huson, Rupp, & Scornavacca 2010: 38) and SUBTREE PRUNING AND REGRAFTING (SPR; Felsenstein 2004: 41–44, Huson, Rupp, & Scornavacca 2010: 38–39). To perform these searches, call the function `optim.parsimony()`. With the argument `rearrangements`, one specifies `"SPR"` or `"NNI"` rearrangements (the former is the default value). NNI and SPR searches can be used after the parsimony ratchet to see if any further optimization of the parsimony score is possible:

```
#SPR optimization
screened.blength.spr <- optim.parsimony(screened.pratchet.rooted,
                                        screened.phydat)
## Final p-score 3612 after  0 nni operations
#Root the tree with Anatolian as the outgroup
screened.spr.rooted <- root(screened.blength.spr,
                            outgroup = anatolian,
                            resolve.root = TRUE)
#Add branch lengths
screened.spr.rooted.blength <- acctran(screened.spr.rooted,
                                       screened.phydat)
```

In this case, optimization was unable to find a better tree. The p-score of the above tree is 3612, which is the same value we obtained above from the branch and bound search. To confirm that the phylogenies are identical, we use `all.equal.phylo()`:

```
#Check if phylogenies are identical
all.equal.phylo(screened.pratchet.blength, screened.spr.rooted.blength)
## [1] TRUE
```

5.6 *Measuring homoplasy and consistency*

There are other measures of tree support besides tree length. Here I introduce two, the CONSISTENCY INDEX and the RETENTION INDEX, both of which provide measures of homoplasy on a tree. HOMOPLASY refers to a situation in which character states develop more than once on a tree. Two types of situations result in homoplasy (Baum & Smith 2013: 93). The first is parallel independent innovation. Changes that are common (e.g., palatalization of velars before front vowels) are good candidates for homoplastic characters. The second type of situation that can result in homoplasy is so-called "Duke of York" changes. To draw again on sound change, a trajectory [a] > [o] > [a] is homoplastic. In the evolutionary biology literature, this phenomenon is known as BACKMUTATION.

A character is CONSISTENT on a given tree if it exhibits the minimum number of changes (i.e., if it shows no homoplasy). The minimum number of changes is always the observed number of character states minus one. For a binary character with values 1 and 0, the minimum number of changes is 1 (i.e., two observed character states minus one). Any tree that accounts for the distribution of the character states 1 and 0 with a single change is consistent with that character. If a tree requires more changes than the minimum, the character is homoplastic on that tree.

The consistency index is a measure of the consistency of a tree:

(2) $CI = \frac{\text{Min}(S)}{S}$

Min(S) is the minimum number of steps required by a tree. As mentioned above, this is equal to the number of observed character states minus one. S is the length of the tree, that is, the actual number of steps on the tree. To calculate the consistency index for a tree, the values of the numerator and denominator are summed for all characters before division. Values of the consistency index range from 1 to close to 0. A consistency index of 1 means that all characters are perfectly consistent on the tree (that is, there is no homoplasy). This situation arises of course when Min(S), the minimum number of changes, equals S, the actual number of changes.

The consistency index is not without its problems (Sanderson & Donoghue 1989, Archie & Felsenstein 1993, Egan 2006: 73). For one, there is a negative correlation between the consistency index and the number of taxa: the consistency index falls as the number of taxa rises (Sanderson & Donoghue 1989). This correlation is explained by the fact that as the number of nodes (i.e., lineage-splitting events) increases, there are more opportunities for homoplasy (Hauser & Boyajian 1997: 97). So with larger datasets, the accuracy of the consistency index is questionable. Second, it is difficult to compare consistency indices across datasets. Third, autapomorphies (unique innovations) and symplesiomorphies (shared inherited traits) both inflate the consistency index, although neither of these situations should affect it since neither involves homoplasy. Finally, the absence of conventions for interpretating consistency indices means that it is not clear what constitutes a high or low value.

The retention index was intended as an improvement on the consistency index (Farris 1989, Lipscomb 1998). Unlike the latter, the former can range from 0 to 1. Like the consistency index, the retention index is the ratio of the observed number of changes and the minimum number of changes, but it is more complex in that it takes into account the maximum number of possible changes. One can think of it as the proportion of the observed number of synapomorphies (i.e., shared innovations) to the maximum possible number of synapomorphies (Egan 2006: 73, Klingenberg & Gidaszewski 2010: 250). It is calculated as follows:

$$(3) \quad RI = \frac{\mathrm{Max}(S)-S}{\mathrm{Max}(S)-\mathrm{Min}(S)}$$

Max(S) is the maximum number of steps required by a tree. To calculate the maximum number of steps on the tree, we count the number of observed states for each character. We select the lowest number in each case and then sum up that value for every character in the dataset.

If the retention index equals one, a character is maximally consistent, i.e., Min$(S) = S$. If the retention index equals zero, a character is maximally homoplastic, i.e., $\mathrm{Max}(S) = S$. (This would mean in addition that the character is parsimony uninformative, i.e., that we cannot use it to make any inferences about the topology of the tree.)

Here are the consistency and retention indices for the trees optimized with nearest neighbor interchange:

```
#Calculate consistency index
formatC(CI(screened.spr.rooted.blength,
           screened.phydat), 3)
## [1] "0.996"
#Calculate retention index
formatC(RI(screened.spr.rooted.blength,
           screened.phydat), 3)
## [1] "0.983"
```

The values of both indices are high, which reflects the fact that the dataset was curated precisely to avoid homoplastic characters.

To see which characters specifically lower the consistency and retention indices, we can use the following code (only the retention index is included here for the sake of space):

```
#Identify indices of characters with a retention index < 1
which(RI(screened.spr.rooted.blength,
         screened.phydat,
         sitewise = TRUE) < 1)
##  [1]   2   3  28  85 103 108 165 166 208 231 244 257 292 293
```

The first two of these are the phonological characters P2 and P3. P2 encodes full “satəm” development, according to which PIE labiovelars merge with velars and “palatals” become affricates or fricatives. P3 refers to the “ruki”-retraction of **s*. The third character is the morphological character M5, which encodes the mediopassive primary marker. The remaining characters are lexical and refer to the following concepts: ‘float2’, ‘head’, ‘ice’, ‘straight’, ‘suck2’, ‘break1’, ‘free’, ‘leave1’, ‘nine’, ‘young2’, and ‘tear’. I refer the reader to the descriptions of the characters by Ringe and Taylor cited above for further discussion.

5.7 *How much phylogenetic structure is in the dataset?*

There is an ongoing debate within Indo-European linguistics over whether the history of the family is in fact best represented by a phylogenetic tree, as opposed to, say, a network. With methods that assign scores to trees (such as parsimony and likelihood methods, the latter of which are presented in section 8 below), we can investigate the degree to which the data exhibit a hierarchical (i.e., tree-like) structure by comparing the optimal tree to trees inferred from permuted datasets. The permuted datasets contain the same number of traits as the real dataset and the same of number of trait values, but their order has been jumbled. For instance the character values 001101 in the original dataset could become 000111 in one of the permuted datasets. I created 100 permuted datasets from the original dataset.

For each dataset, I inferred a phylogeny using the parsimony ratchet and recorded the length of each tree (i.e., the sum of all the branch lengths). I then compared the lengths of these one hundred trees to that of the tree from the original dataset. In effect, this is a comparison between the tree inferred from the real dataset to one hundred trees from random data (otherwise known as a PERMUTATION TAIL PROBABILITY test). If the length of the tree inferred from the real data differs from the lengths of the trees inferred from the randomized datasets, the data are said to contain more tree-like structure than would be expected from random data (Baum & Smith 2013: 268).

The following plot reveals that the length of the parsimony ratchet tree from the original dataset is considerably lower than that of all the trees inferred from the permuted dataset:

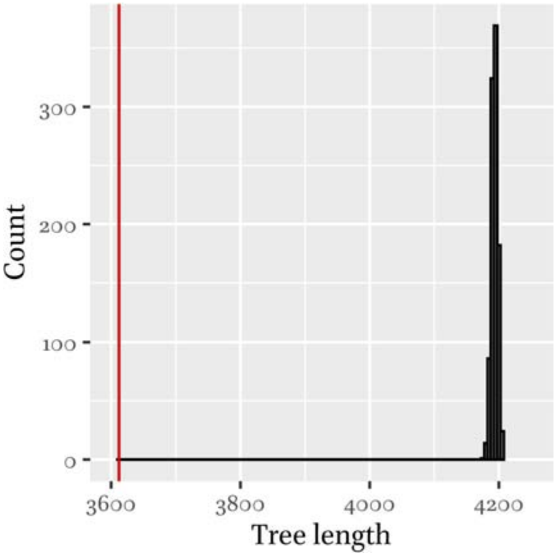


FIGURE 9
Length of the parsimony ratchet tree compared to the length of trees inferred from permuted datasets

The red line represents the length of the tree inferred from the original dataset (3612) and the black bars the lengths of the trees inferred from the permuted datasets. The results of the permutation tail probability test do not of course mean that Indo-European needs to be modeled with a tree. It means that this specific dataset contains more phylogenetic signal than one would expect from random data. Given that most of the phonological characters define clades (Nakhleh, Ringe, & Warnow 2005: 394), the result is perhaps not surprising. With a different Indo-European dataset, one might obtain different results.

5.8 *Issues*

Although parsimony methods are closest in spirit to traditional subgrouping methods and yield good results, they are not without their pitfalls. For one, the assumption that the true tree is characterized by the fewest number of changes may be inappropriate for some data sets (see, e.g., Penzl 1960: 216). Application of Ockham's razor to the vicissitudes of history can dupe us into believing that linguistic history is tidier and more economical than it actually is (see, e.g., Sober 1988 and Sober 2015 for discussion of the methodological and philosophical issues of parsimony). In datasets where characters have undergone a number of changes with the result that multiple taxa exhibit the same states, two interrelated problems arise (Swofford et al. 1996: 427, Schulmeister 2004, Bergsten 2005, Baum & Smith 2013: 205–07, Yang 2014: 99–100, Warnow 2018: 161–64). First, maximum parsimony methods underestimate the amount of change. Second, since the methods are designed to minimize homoplasy, shared character traits will be treated as synapomorphies. In other words, if two taxa have independently undergone a lot of change (i.e., have long branches), maximum parsimony will interpret the changes as shared innovations and pair them together. Felsenstein (1978b) called attention to this problem in the context of DNA sequences. He referred to it as LONG BRANCH ATTRACTION, although the problem also arises in trees with equal branch lengths. Maximum parsimony is therefore said to be POSITIVELY MISLEADING (Warnow 2018: 161). We typically expect an estimate to improve with more data. This is known as statistical consistency (Warnow 2018: 146). Rather than converge to the true tree as the amount of data increases, maximum parsimony methods can converge to the wrong tree.

Linguistically, the weaknesses of parsimony methods are especially salient when it comes to phylogenetic inference from sound change. It is well known that certain types of sound changes are more common than others (e.g., Garrett & Johnson 2013: 52). Given enough time, it is likely languages will individually undergo such sound changes. Such a homoplastic scenario would be interpreted by the maximum parsimony algorithms as evidence for shared innovation. We should therefore get the best results from maximum parsimony methods with datasets characterized by fewer transitions (cf. Baum & Smith 2013: 187). This is one reason why maximum parsimony methods may be of greater utility for linguistic phylogenetics than for evolutionary biology, since linguistic datasets are far more restricted in the time depth of their characters. At shallower time depths, there is less opportunity for change and long branch attraction.

6 Measuring clade support

Once our phylogenetic method infers a tree, we need to ask ourselves how much confidence we should have that the estimated tree represents the true tree. Node support is a measure of the extent to which the data support the clades in the phylogeny. The most widely used measure is the nonparametric bootstrap (Baum & Smith 2013: 273), which was first introduced into phylogenetic analysis by Felsenstein (1985) (see further Sanderson 1989, Sanderson 1995, Efron, Halloran, & Holmes 1996, Egan 2006, Huson, Rupp, & Scornavacca 2010: 43–44). The basic idea is to assess the degree to which our sample character data approximate the true phylogeny. Bootstrap analysis creates other possible datasets by randomly sampling from the original dataset with replacement (Efron 1979, Efron & Tibshirani 1993, Efron 2003).

6.1 *Bootstrapping*

The basic procedure is as follows (Durbin et al. 1998: 180). For a dataset with *n* characters, randomly sample the dataset *n* times with replacement. These datasets are known as PSEUDOREPLICATES. Sampling with replacement will yield pseudoreplicates in which some characters are represented more than once, while some characters are not represented at all. Below I create 100 bootstrapped datasets and apply the method under discussion to each. For each clade inferred from the original dataset, the bootstrap function then tallies the number of bootstrapped datasets that contain that clade. Dividing this number by the total number of bootstrapped datasets yields the confidence value for a particular clade. In short, we are using the character data itself to infer how reliable our estimated phylogeny is. It is hard to overestimate the importance of measuring clade support in Indo-European phylogenetics. It is absolutely critical that we know how robust our results are.

Bootstrap analysis can be carried out with the `boot.phylo()` function from `ape` (for more on bootstrap analysis in R, see Paradis 2012: 174–79).[26] We begin by setting a seed:

[26] Bootstrap analyses can also be carried out with `bootstrap.phyDat()` and `plotBS()` in `phangorn`.

```
#Seed for replication
set.seed(233)
```

By using the `set.seed()` function, we essentially assign a particular sequence of random samples an index. This then enables one to replicate the results of the bootstrap sample. In other words, calling `set.seed(233)` will ensure that the same set of pseudoreplicates is generated each time. (The value 233 has no significance; it is simply the starting point of the pseudo-random number generator.) For more on random seeds, call `?set.seed`.

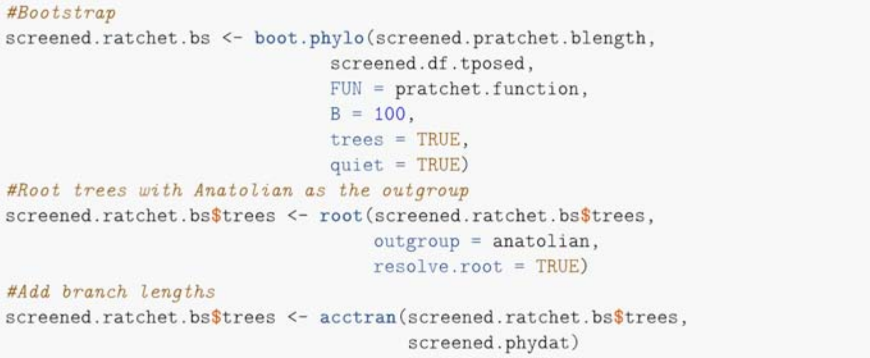
We then write a function for the phylogenetic analysis of our bootstrapped samples:

```
#Determine the method for inferring trees
pratchet.function <- function (x) {pratchet(phyDat(x,
                                   type = "USER",
                                   levels = screened.codings))}
```

This function calls the parsimony ratchet on the input dataset and will then root the output with Anatolian as an outgroup. The bootstrap function `boot.phylo()` also requires a dataset with taxa (i.e., languages) as rows and characters as columns. (In the `screened.df` dataset, the taxa are columns and the characters are rows.) We transpose the dataframe as follows:

```
#Transpose dataframe
screened.df.tposed <- t(screened.df)
```

Our transposed dataset and phylogenetic function will then serve as arguments of the function `boot.phylo()` from the ape package, with which we run the bootstrap analysis:



```
#Bootstrap
screened.ratchet.bs <- boot.phylo(screened.pratchet.blength,
                              screened.df.tposed,
                              FUN = pratchet.function,
                              B = 100,
                              trees = TRUE,
                              quiet = TRUE)
#Root trees with Anatolian as the outgroup
screened.ratchet.bs$trees <- root(screened.ratchet.bs$trees,
                                  outgroup = anatolian,
                                  resolve.root = TRUE)
#Add branch lengths
screened.ratchet.bs$trees <- acctran(screened.ratchet.bs$trees,
                                     screened.phydat)
```

The argument `B = 100` sets the number of bootstrap replicates at 100, while `trees = TRUE` keeps all the trees from the analysis and `rooted = TRUE` specifies that the trees should be treated as rooted.

Once we have our bootstrap trees we get the scores for each clade as follows:

```
#Bootstrap scores
scores.ratchet.bs <- prop.clades(screened.pratchet.blength,
                                 screened.ratchet.bs$trees,
                                 rooted = TRUE)
#Convert into a dataframe for edge width
scores.df <- as.data.frame(scores.ratchet.bs)
names(scores.df) <- "support"
tips <- as.data.frame(rep(100, 24))
names(tips) <- "support"
support <- rbind(tips, scores.df)
support <- as.vector(support$support)
```

The function `prop.clades()` tallies the frequency of the bipartitions in a given phylogenetic tree (here `screened.pratchet.blength`) among the bootstrap trees. This is the ratchet parsimony tree annotated with bootstrap scores (the thickness of the branches also reflects these scores):

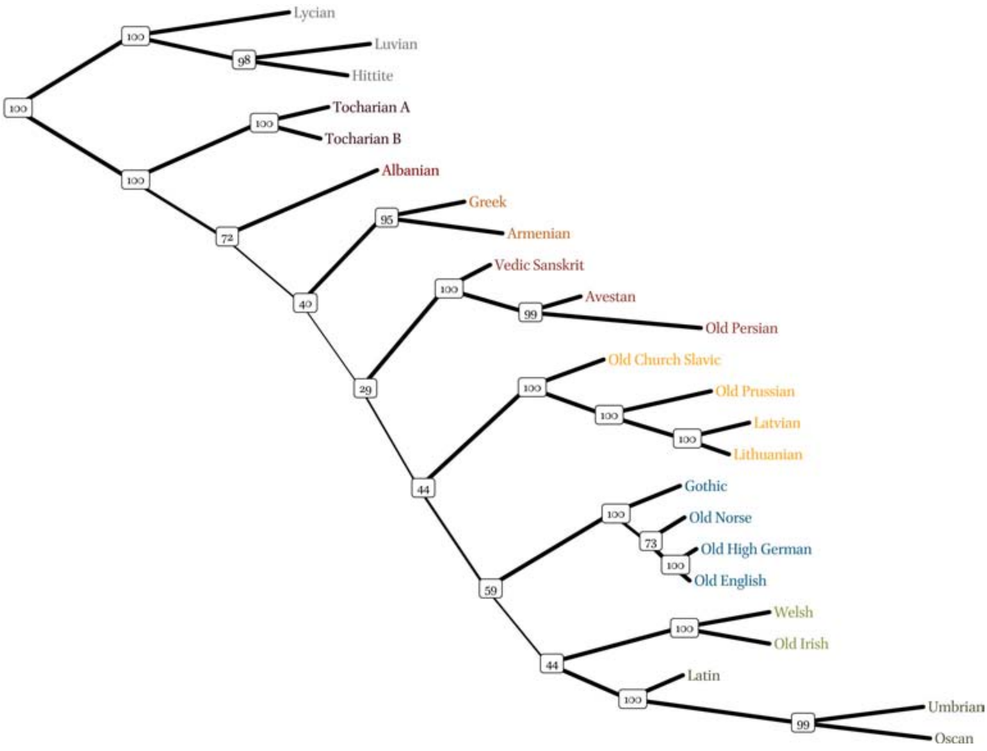


FIGURE 10 Parsimony ratchet tree with bootstrap scores

The bootstrap scores indicate how many times a set of languages formed a clade among the total number of trees inferred from the pseudoreplicate datasets. For instance, the Italic, Celtic, and Germanic languages form a clade in fifty-nine of the 100 pseudoreplicates. In other words, the data provide only weak support for such a clade. By contrast, every pseudoreplicate contains a clade with Latin, Oscan, and Umbrian.

It is critical that one interpret bootstrap scores accurately. First off, a high bootstrap score does not corroborate the existence of a particular clade (cf. Nichols & Warnow 2008: 773). Rather, it means that the dataset in question offers robust support for such a clade. Likewise, a low bootstrap score should not be interpreted to mean that a particular clade did not exist. It simply means that the data from which the bootstrap trees were inferred do not support such a clade. Examination of further data could either corroborate or disconfirm the existence of such a clade (cf. Egan 2006: 80). Ewens & Grant (2005: 525) point out in addition that if the assumptions of an estimation procedure are at odds with the true history, then any error in the estimated tree will tend to be shared with the trees in the pseudoreplicates.

The results in the tree above are sobering and largely recapitulate what has been known at least since Brugmann (1884: 226), namely that the innovations that define late diverging clades such as, e.g., Indo-Iranian are clear, but innovations that define earlier diverging clades, such as the ancestor of Germanic, Celtic, and Italic, are scarce (Garrett 1999: 147). The Tocharian languages and Albanian are an exception to this trend in the tree above, as their position is more robust. These languages aside, the takeaway message from the bootstrap scores in the above phylogeny is that the evidence for the post-Tocharian clades in the dataset is weak and that we need to find data with a more robust phylogenetic signal.

If one removes the homoplastic characters mentioned in section 5.6 above, the bootstrap scores improve somewhat:

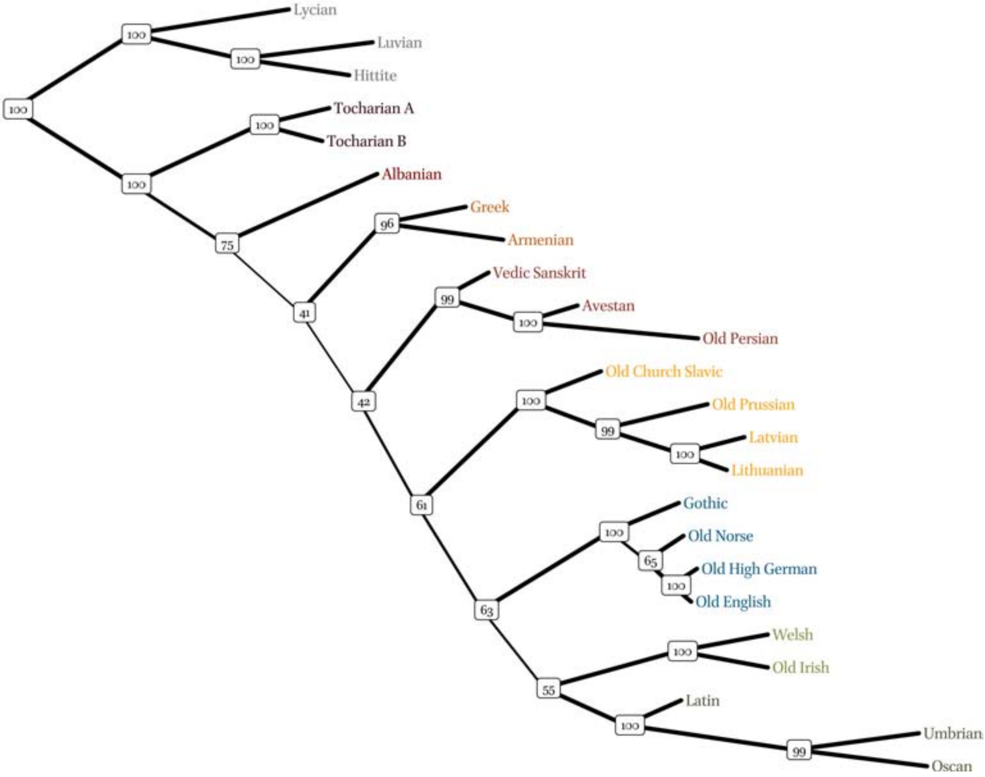


FIGURE 11 Parsimony ratchet tree with bootstrap scores (pruned data set)

The data now provide better support for a clade containing Balto-Slavic, Germanic, Celtic, and Italic.

We can calculate consensus and maximum clade credibility trees for the bootstrap trees, just as we did for the branch and bound trees in sections 5.3 and 5.4 above:

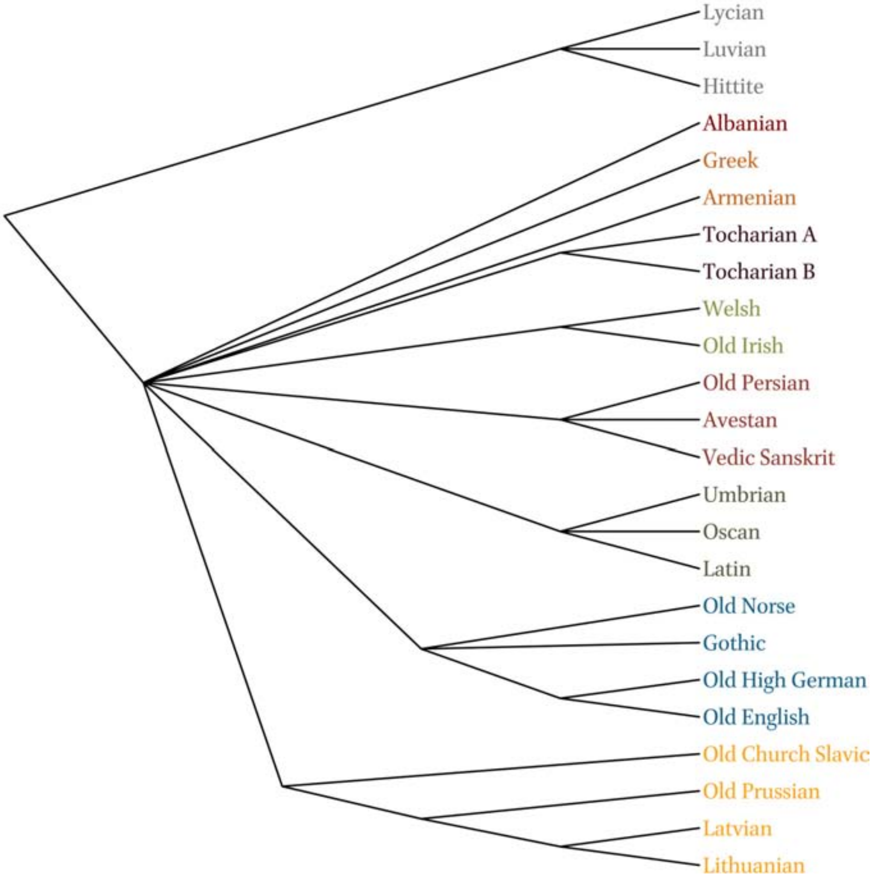


FIGURE 12 Strict consensus tree

The multifurcation after the departure of the Anatolian languages reflects the fact that there is no consensus among the bootstrap trees concerning the early topology of Proto-Nuclear-Indo-European.

To calculate a majority-rule consensus tree, use the argument `p = 0.5`.

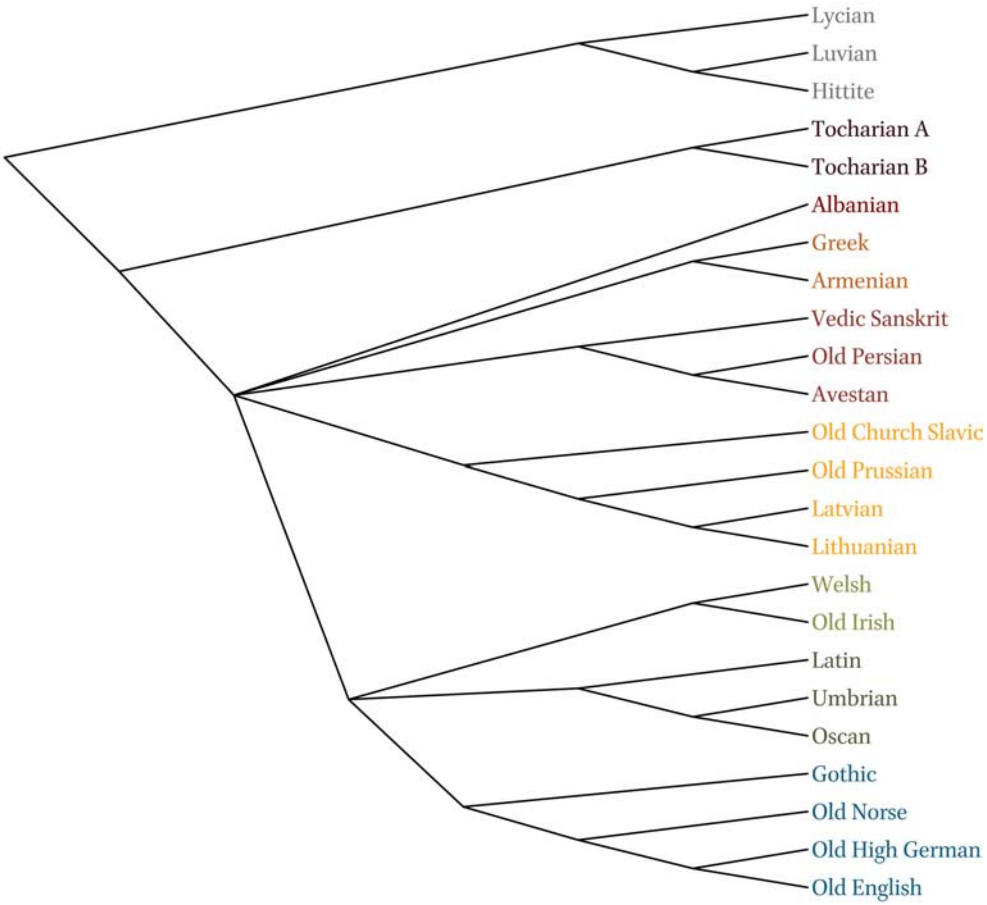


FIGURE 13 Majority-rule consensus tree

It is interesting that Proto-Tocharian is now a sister to the ancestor of the remaining archaic Indo-European languages, since this was not the case in the strict consensus tree.

The maximum clade credibility tree looks as follows:

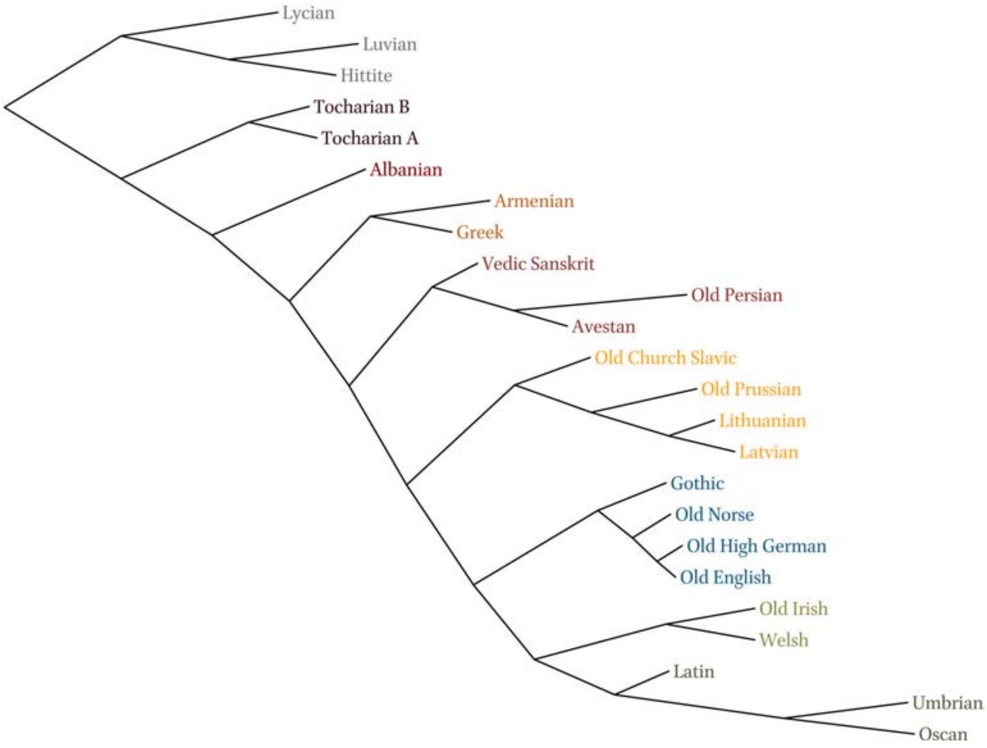


FIGURE 14 Maximum clade credibility tree

In contrast to the two consensus trees above, the maximum clade credibility tree is one of the trees from the bootstrapped pseudoreplicates.

6.2 *Issues*

The use of the bootstrap in phylogenetic analysis is not without its problems. Egan (2006: 75–76) outlines the most important ones. Underpinning bootstrap analysis is the assumption that a large number of characters have been sampled randomly from a population of uncorrelated characters that have the same distributions. The use of the bootstrap above violates these assumptions (see further Sanderson 1989: 115–16). First, the character data was not chosen randomly. It was in fact heavily curated. Second, it is far from clear that the character data is independent and identically distributed (i.i.d.). At this point, we simply do not know that extent to which values of one character may have influenced those of another.

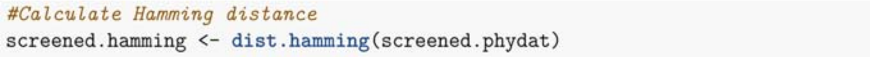
7 Distance-based methods

In this section, I introduce two distance-based methods for phylogenetic inference (Sneath & Sokal 1973, Aldenderfer & Blashfield 1984, Van de Peer 2009, Everitt et al. 2011). The crucial distinction between optimality methods (such as maximum parsimony and maximum likelihood) and distance-based methods is that the latter infer phylogenies by applying an algorithm to a distance matrix. The distance matrix is a measure of the dissimilarity between each taxon in the dataset, which is created from the original data.

7.1 *Hamming distance*

HAMMING DISTANCE (also known as DEGREE OF DIVERGENCE and PAIRWISE DISTANCE) is a simple measure of dissimilarity: it is the number of characters for which two languages differ (i.e., exhibit non-identical character values) divided by the total number of characters (Graur & Li 2000: 74, Baum & Smith 2013: 232–34). If two languages differ at 5 out of 20 sites then they have a Hamming distance of 0.25.

Hamming distance is calculated with the function `dist.hamming()` from the `phangorn` package:[27]



```
#Calculate Hamming distance
screened.hamming <- dist.hamming(screened.phydat)
```

This distance matrix will be the input to the UPGMA and NJ algorithms presented in the subsequent sections.

7.2 *Unweighted Pair Group Method with Arithmetic Mean* (*UPGMA*)

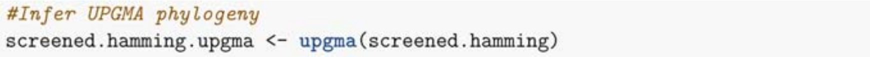
The UPGMA algorithm, created by Sokal & Michener (1958), works as follows. (Swofford et al. 1996: 488–90, Johnson 2008: 182–214, Levshina 2015: 310–11 and Kassambara 2017 provide instruction for implementing distance-based methods in R; for a UPGMA analysis of linguistic data, see Delmestri & Cristianini 2010.) The two languages in a distance matrix that are least dissimilar (e.g., have the lowest hamming distance) are paired together under a node. The lengths of the branches from the tips to the node are then calculated. Once

[27] The Hamming dissimilarity measure is crude, because it assumes that the pairwise distance between any two languages is tantamount to the amount of change that they have undergone. More sophisticated measures of calculating the distance between two languages are based on explicit evolutionary models, e.g., maximum likelihood distance (see Yang 2014: 17–22, 27–33). One can calculate maximum likelihood distance in `phangorn` with the function `dist.ml()`.

this first cluster with its two languages has been created, the algorithm returns to the distance matrix and replaces the row and column in which these two languages appeared with their cluster. The distance between this cluster and the remaining taxa in the distance matrix is then calculated. Once the distance matrix has been updated with the new distances, the process is repeated. That is, the two least dissimilar taxa are paired together in a cluster, their branch lengths are calculated, and the distances in the matrix are updated. The algorithm concludes once there is only one item remaining in the distance matrix.

The most salient property of the UPGMA algorithm is that it assumes an equal rate of change across all branches. As a result, UPGMA trees are ULTRAMETRIC, which means that the distance from the root to each tip is equal.

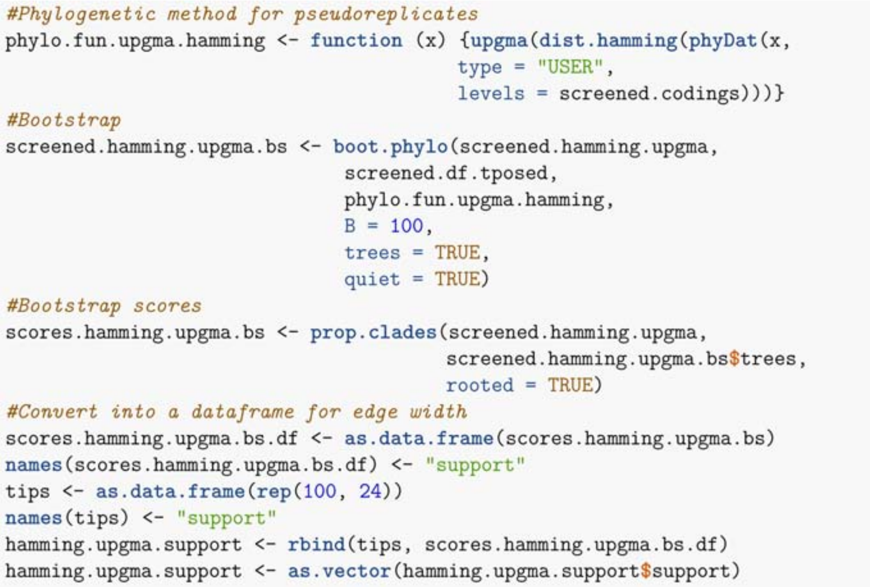
Since the UPGMA algorithm creates rooted trees with branch lengths, inferring a UPGMA tree is as simple as calling the function upgma() on the distance matrix:



```
#Infer UPGMA phylogeny
screened.hamming.upgma <- upgma(screened.hamming)
```

To view the distance matrix, type screened.hamming.upgma into the console. The matrix is large, so I do not present it here. The object screened.hamming.upgma will be the input to the UPGMA and NJ algorithms presented below.

Bootstrap analysis of the UPGMA tree based on Hamming distances is carried out as follows:



```
#Phylogenetic method for pseudoreplicates
phylo.fun.upgma.hamming <- function (x) {upgma(dist.hamming(phyDat(x,
                                          type = "USER",
                                          levels = screened.codings)))}
#Bootstrap
screened.hamming.upgma.bs <- boot.phylo(screened.hamming.upgma,
                              screened.df.tposed,
                              phylo.fun.upgma.hamming,
                              B = 100,
                              trees = TRUE,
                              quiet = TRUE)
#Bootstrap scores
scores.hamming.upgma.bs <- prop.clades(screened.hamming.upgma,
                                       screened.hamming.upgma.bs$trees,
                                       rooted = TRUE)
#Convert into a dataframe for edge width
scores.hamming.upgma.bs.df <- as.data.frame(scores.hamming.upgma.bs)
names(scores.hamming.upgma.bs.df) <- "support"
tips <- as.data.frame(rep(100, 24))
names(tips) <- "support"
hamming.upgma.support <- rbind(tips, scores.hamming.upgma.bs.df)
hamming.upgma.support <- as.vector(hamming.upgma.support$support)
```

Here is the UPGMA tree annotated with bootstrap scores:

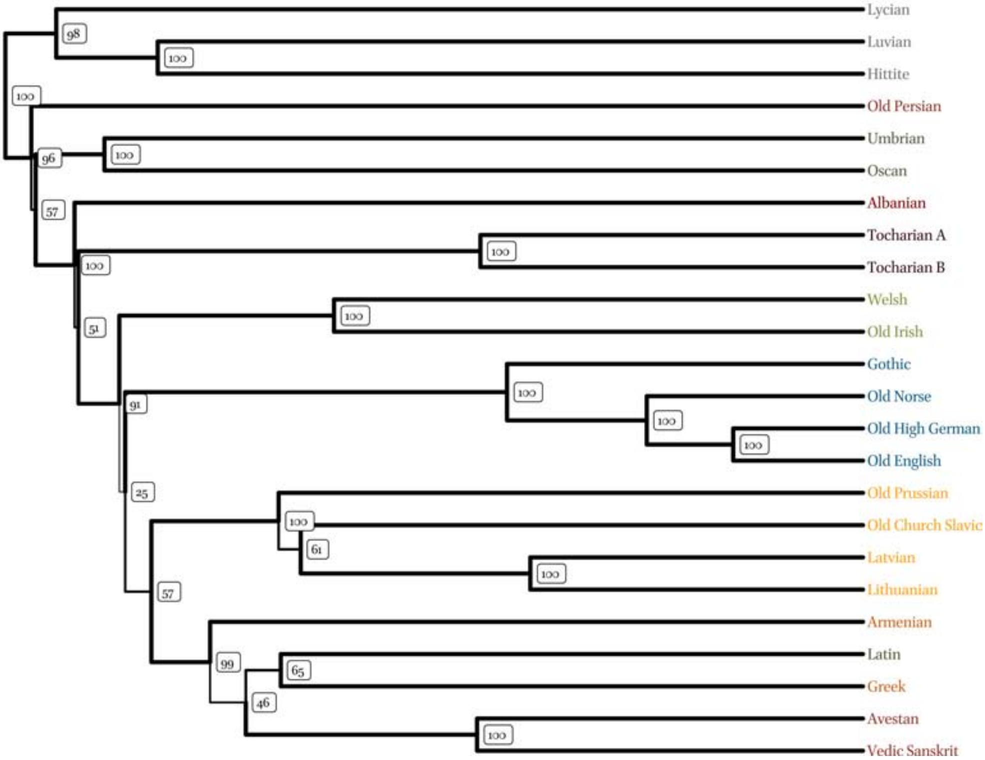


FIGURE 15 UPGMA tree (Hamming distance)

This tree suffers from a number of glaring problems (cf. Barbançon et al. 2013: 143, 161, 163–64). First, it fails to establish an immediate common ancestor for Old Persian, Avestan, and Vedic Sanskrit. Oscan and Umbrian should have been paired with Latin, which instead forms a clade with Greek. It is also peculiar that Old Church Slavic diverges after Old Prussian but before Lithuanian and Latvian.

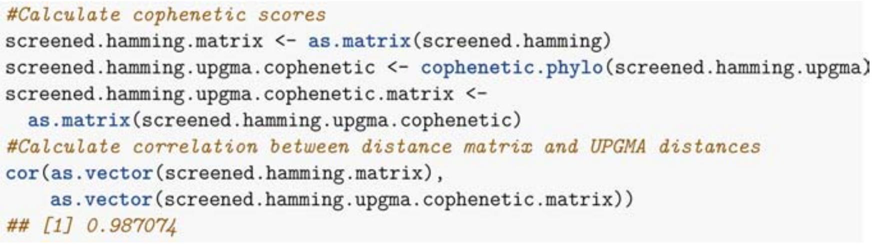
The consistency and retention indices for the UPGMA tree are as follows:

```
#Calculate consistency index
CI(screened.hamming.upgma, screened.phydat)
## [1] 0.9761259
#Calculate retention index
RI(screened.hamming.upgma, screened.phydat)
## [1] 0.8918919
```

The retention indices are lower on the UPGMA tree compared to the maximum parsimony trees presented in section 5.6 above, which means that there is more homoplasy on this tree.

7.2.1 Cluster validation

The creation of a phylogenetic tree from a distance matrix inevitably involves the loss of information. The amount of information lost varies according to the clustering algorithm. To evaluate the differences between the pairwise distances of the distance matrix and the distances between taxa in the tree, we rely on measures of cophenetic distance (Kassambara 2017: 73). The function `cophenetic.phylo()` in `ape` computes the pairwise distances between the pairs of tips from a phylogenetic tree using its branch lengths (see further Paradis 2012: 125–33). We then run `cor()` on both distance vectors to assess the correlation between the original distances and the cophenetic distance:



```
#Calculate cophenetic scores
screened.hamming.matrix <- as.matrix(screened.hamming)
screened.hamming.upgma.cophenetic <- cophenetic.phylo(screened.hamming.upgma)
screened.hamming.upgma.cophenetic.matrix <-
  as.matrix(screened.hamming.upgma.cophenetic)
#Calculate correlation between distance matrix and UPGMA distances
cor(as.vector(screened.hamming.matrix),
    as.vector(screened.hamming.upgma.cophenetic.matrix))
## [1] 0.987074
```

The closer the correlation coefficient is to 1, the more accurately the clustering solution reflects the data. Values above 0.75 are considered good in some fields. It is not yet clear what constitutes a reliable value for linguistic phylogenetics.

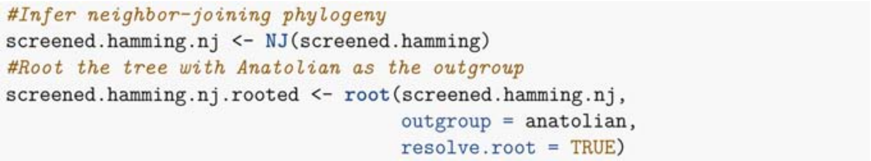
7.2.2 Issues

The central weakness of UPGMA is that its results will be wide of the mark if the rates of change are not equal among the languages in the dataset (Felsenstein 2004: 165). As the tree above makes clear, UPGMA does not yield good results for our Indo-European data. (Nakhleh et al. 2005: 182, 185–86 and Barbançon et al. 2013: 166 came to a similar conclusion.) UPGMA has been heavily criticized for its assumption of ultrametricity, but it is crucial to understand that the nature of the tree that the alogorithm produces is fundamentally different from that of either parsimony or likelihood methods. UPGMA tells us how similar pairs of languages are at the synchronic stages represented by the data. Given that the Indo-European phylogeny is not ultrametric, it may make more sense to interpret the UPGMA tree above not as a model of linguistic history, but rather as a model of similarity. Viewed thus, it actually makes a fair amount of sense that Latin and Greek form a clade. It is less clear how one motivates the position of Old Persian, however.

7.3 *Neighbor Joining*

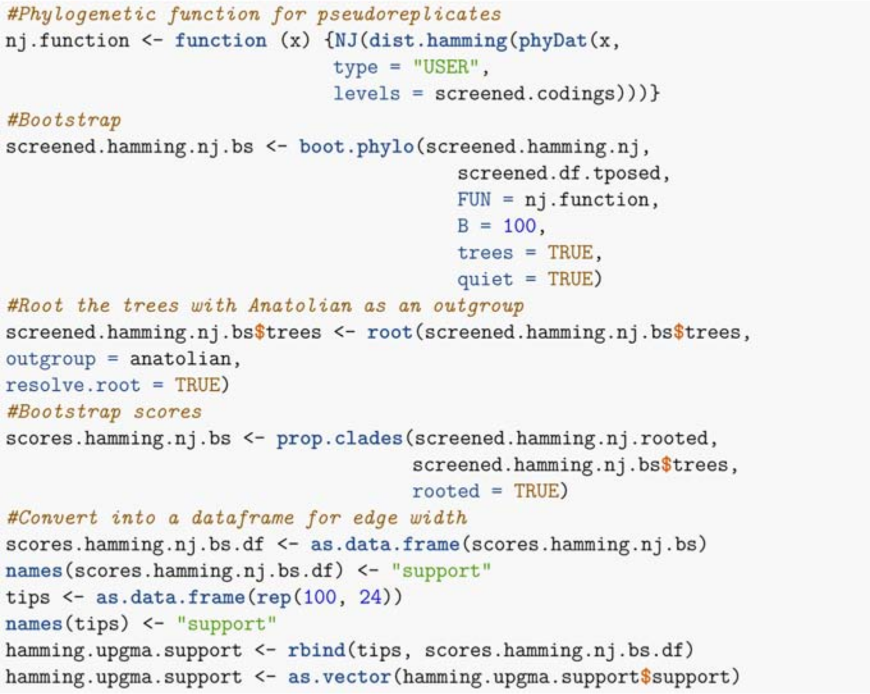
The neighbor-joining algorithm (Saitou & Nei 1987) is similar in spirit to that of UPGMA in that it estimates a phylogeny from a distance matrix. In its details, however, the algorithm works quite differently. For one, it is a divisive algorithm that begins with a star tree (Swofford et al. 1996: 488, Yang 2014: 92). Nodes in the tree are constructed not from the distance matrix itself, but rather from a modified distance matrix. This distance matrix adjusts the distance between each pair of taxa on the basis of their average divergence from all other nodes (Swofford et al. 1996: 488). Neighbor joining is guaranteed to recover the true tree if the distance matrix happens to be an exact reflection of the tree (Felsenstein 2004: 166). In contrast to UPGMA, Neighbor joining does not assume that the tree is ultrametric, that is, the taxa are not assumed to have all diverged to an equal extent (Baum & Smith 2013: 234–36, Swofford et al. 1996: 488). Neighbor joining is often used to infer a start tree for other methods, such as Maximum Likelihood Estimation (Felsenstein 2004: 169, Swofford et al. 1996: 490; MLE is presented in section 8 below), not least because it can be used with up to hundreds of taxa (Felsenstein 2004: 166).

We infer a neighbor-joining tree from our distance matrix with the command `NJ()`:



```r
#Infer neighbor-joining phylogeny
screened.hamming.nj <- NJ(screened.hamming)
#Root the tree with Anatolian as the outgroup
screened.hamming.nj.rooted <- root(screened.hamming.nj,
                                   outgroup = anatolian,
                                   resolve.root = TRUE)
```

Although the neighbor-joining algorithm produces trees with branch lengths, the trees have to be rooted, which is why I called the `root()` function in the above snippet. The following code then performs the bootstrap analysis:



```
#Phylogenetic function for pseudoreplicates
nj.function <- function (x) {NJ(dist.hamming(phyDat(x,
                             type = "USER",
                             levels = screened.codings)))}
#Bootstrap
screened.hamming.nj.bs <- boot.phylo(screened.hamming.nj,
                                     screened.df.tposed,
                                     FUN = nj.function,
                                     B = 100,
                                     trees = TRUE,
                                     quiet = TRUE)
#Root the trees with Anatolian as an outgroup
screened.hamming.nj.bs$trees <- root(screened.hamming.nj.bs$trees,
outgroup = anatolian,
resolve.root = TRUE)
#Bootstrap scores
scores.hamming.nj.bs <- prop.clades(screened.hamming.nj.rooted,
                                    screened.hamming.nj.bs$trees,
                                    rooted = TRUE)
#Convert into a dataframe for edge width
scores.hamming.nj.bs.df <- as.data.frame(scores.hamming.nj.bs)
names(scores.hamming.nj.bs.df) <- "support"
tips <- as.data.frame(rep(100, 24))
names(tips) <- "support"
hamming.upgma.support <- rbind(tips, scores.hamming.nj.bs.df)
hamming.upgma.support <- as.vector(hamming.upgma.support$support)
```

The neighbor-joining tree with annotated bootstrap scores looks as follows:

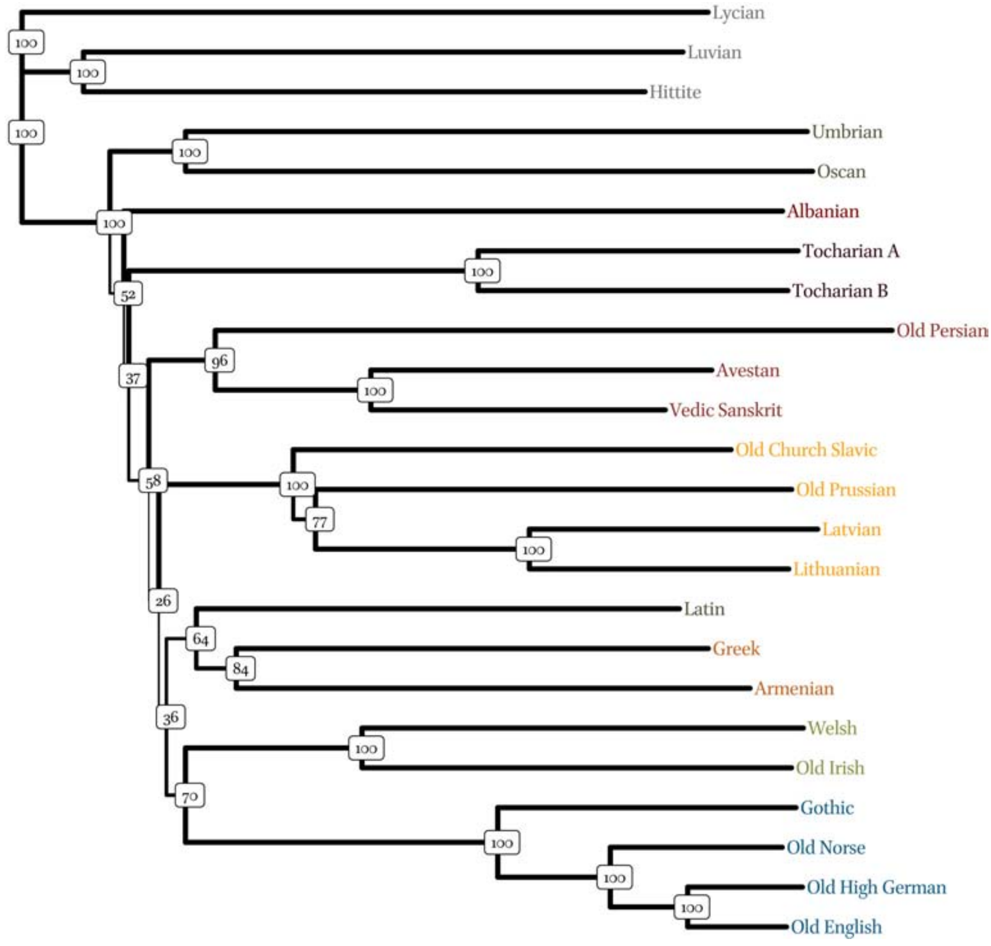


FIGURE 16 NJ tree (Hamming distance)

It is interesting that the neighbor-joining algorithm stumbles with the position of Latin, just as the UPGMA algorithm did. This time, however, Latin is assigned to a clade with Greco-Armenian. All taxa are otherwise assigned to the correct clades. Branch lengths reflect the amount of lingusitic change on a particular lineage, so according to this tree Old Persian is the most innovative of the languages in the dataset, and Hittite is the most conservative.

Here are the measures of homoplasy for the NJ tree:

```
#Calculate consistency index
CI(screened.hamming.nj.rooted, screened.phydat)
## [1] 0.9876475
#Calculate retention index
RI(screened.hamming.nj.rooted, screened.phydat)
## [1] 0.9447174
```

The consistency and retention indices suggest low levels of homoplasy on the tree. The retention index is noticeably higher for the NJ tree compared to the UPGMA tree.

Finally, the following code carries out cluster validation:

```
#Calculate cophenetic scores
screened.hamming.matrix <- as.matrix(screened.hamming)
screened.hamming.nj.rooted.cophenetic <-
  cophenetic.phylo(screened.hamming.nj.rooted)
screened.hamming.nj.rooted.cophenetic.matrix <-
  as.matrix(screened.hamming.nj.rooted.cophenetic)
#Calculate correlation between distance matrix and NJ distances
cor(as.vector(screened.hamming.matrix),
    as.vector(screened.hamming.nj.rooted.cophenetic.matrix))
## [1] 0.9949778
```

The correlation between the distance matrix and the branch lengths of the phylogenetic tree is high.

7.3.1 Issues

One of the challenges of distance-based algorithms is overabundance. There are not only many different ways of calculating the distance between languages, but also many different ways of constructing a phylogenetic tree from distance matrices (Everitt et al. 2011: 80–83). Which methods are best for linguistic phylogenetics is not always so clear. All that seems certain at this point is that algorithms that assume ultrametricity seem implausible for linguistic datasets. Under distance-based methods, symplesiomorphies (shared archaisms) are not distinguished from synapomorphies (shared innovations): they both increase the proximity of taxa (Bowern 2017: 424).

8 Maximum likelihood

With the publication of the seminal Felsenstein (1981), phylogenetic inference takes a statistical turn (Oaks 2015: 1122; for the earlier history of maximum likelihood, see Huelsenbeck & Crandall 1997: 441). Phylogenetic estimation begins to be viewed as a problem of statistical inference whereby characters (or molecular sequences) evolve along the paths of a phylogeny via probabilistic processes (Yang 2014: vii, Warnow 2018: 145). The central question in maximum likelihood estimation is the following: What tree (and model of character change) maximizes the probability of the observed data? The probability of an observed dataset given a particular tree and model of evolution is known as the LIKELIHOOD, a statistical concept of fundamental importance (see, e.g., Pawitan 2001). Like maximum parsimony, phylogenetic inference in maximum likelihood relies on an optimality criterion (for a comparison of maximum likelihood and maximum parsimony, see Lewis 1998): the best estimate of the phylogeny is the one that maximizes the likelihood of the observed data (Felsenstein 1981, Baum & Smith 2013: 240).

8.1 *Likelihood and maximum likelihood*

Before explicating maximum likelihood in a phylogenetic context, I first illustrate the concept of likelihood with the well-worn but still useful example of fair and biased coins. The probability that a fair coin will land on heads (or tails) is 0.5. In other words, given that the coin is fair, the probability of obtaining heads is equal to the probability of obtaining tails. So if we toss a coin one hundred times, we expect approximately fifty heads. If one hundred people each toss a coin one hundred times, we expect a distribution such as the following:

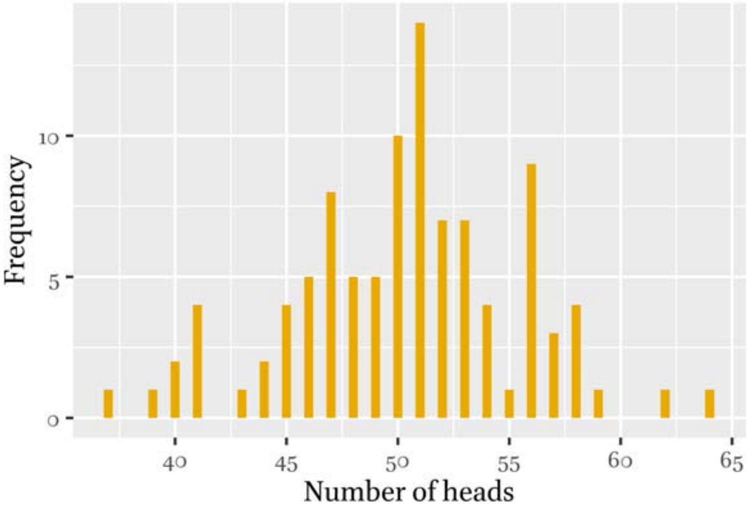


FIGURE 17 Distribution of heads for a fair coin

Nearly all participants get between forty and sixty heads, with results around fifty being the most common.

Now imagine that we ask one hundred people to each flip a coin one hundred times, but end up with starkly different results:

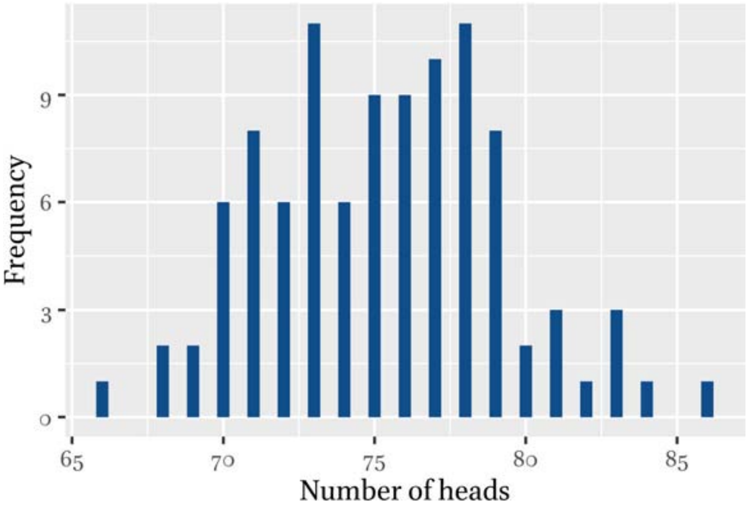
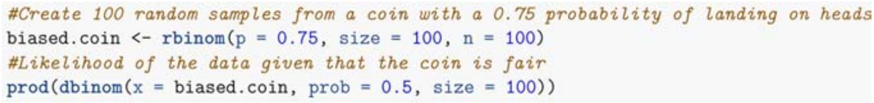


FIGURE 18 Distribution of heads for a biased coin

Most people now get between seventy and eighty heads, a far cry from the previous distribution. It is hard to imagine that the coins would so frequently land on heads if the probability of doing so were really 0.5. What then is the probability that this coin will land on heads? In other words, what probability of obtaining heads makes the observed data above most likely? This question lies at the heart of maximum likelihood phylogenetic estimation. We want to know the probability of the observed data given a particular model of the coin, i.e., a particular probabilty of obtaining heads. The parameter value that makes the observed number of heads the most likely will then be our optimal model.

We calculate the likelihood of the biased distribution above according to different parameter values as follows (I omit the details of this calculation in the interest of simplifying the discussion):



```
#Create 100 random samples from a coin with a 0.75 probability of landing on heads
biased.coin <- rbinom(p = 0.75, size = 100, n = 100)
#Likelihood of the data given that the coin is fair
prod(dbinom(x = biased.coin, prob = 0.5, size = 100))
```

```
## [1] 0
```

```
#Likelihood of the data given 65% probability of heads
prod(dbinom(x = biased.coin, prob = 0.65, size = 100))
```

```
## [1] 1.298017e-235
```

```
#Likelihood of the data given 75% probability of heads
prod(dbinom(x = biased.coin, prob = 0.75, size = 100))
```

```
## [1] 3.692512e-128
```

```
#Likelihood of the data given 85% probability of heads
prod(dbinom(x = biased.coin, prob = 0.85, size = 100))
```

```
## [1] 2.927659e-266
```

These numbers are extraordinaly small (in fact, the first number is not actually 0, but it is so infinitesimal that R is representing it as 0). For this reason, it is customary to work with the log-likelihood, that is, the natural logarithm of the likelihood values:

```
#Log-likelihood of the data given that the coin is fair
sum(dbinom(x = biased.coin, prob = 0.5, size = 100, log = T))
```

```
## [1] -1636.701
```

```
#Log-likelihood of the data given 65% probability of heads
sum(dbinom(x = biased.coin, prob = 0.65, size = 100, log = T))
```

```
## [1] -540.8467
```

```
#Log-likelihood of the data given 75% probability of heads
sum(dbinom(x = biased.coin, prob = 0.75, size = 100, log = T))
```

```
## [1] -293.4246
```

```
#Log-likelihood of the data given 85% probability of heads
sum(dbinom(x = biased.coin, prob = 0.85, size = 100, log = T))
```

```
## [1] -611.4134
```

These transformed probabilities are known as log-likelihood scores. Values closer to zero represent higher log-likelihoods.[28] Working with log-likelihoods makes it easier to see that the likelihood increases between 0.5 and 0.75 but

[28] Since probabilities lie between 0 and 1, log-likelihood scores have a maximum of 0, but in practice will always be negative.

then drops again with 0.85. This suggests that the maximum likelihood estimate of the biased coin is around 0.75.

The following plot presents the log-likelihoods for a range of parameter values:

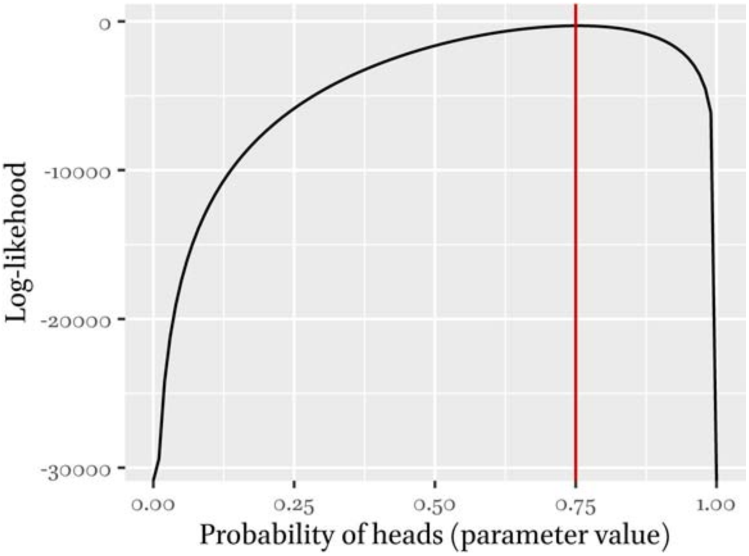


FIGURE 19 Maximum likelihood estimate

The graph plots log-likelihood values as a function of probabilities. The log-likelihood reaches its maximum value (−293) when the probability of heads is 0.75. The maximum likelihood estimate of the probability of obtaining heads with our biased coin is therefore 0.75. In the next section, we apply this reasoning to phylogenetic inference.

8.2 *Maximum likelihood in a phylogenetic context*

As noted above, maximum likelihood phylogenetic estimation identifies the best tree(s) on the basis of an optimality criterion (for general introductions, see Huelsenbeck & Crandall 1997, Felsenstein 2004: 248–74, Schmidt & von Haeseler 2009, Huson, Rupp, & Scornavacca 2010: 40–43, Baum & Smith 2013: 238–47, Yang 2014: 102–51).[29] The optimal tree (or trees) is the one that makes the observed data most likely.

[29] For a range of studies, both linguistic and cultural, that rely on likelihood methods, see (Pagel 2017: 153). Pagel (2000) is an application of maximum likelihood methods to Indo-European specifically.

To illustrate the use of maximum likelihood in a phylogenetic context, imagine a binary linguistic character X with values 0 and 1 in three languages. Language A exhibits the value 0; language B, the value 1; and language C, the value 0. Altogether we therefore have the sequence 010. We begin by calculating the probability of this sequence given the following tree (in order to keep this example simple, I stipulate ancestral states and only take account of tree topology and not other parameters such as branch length):

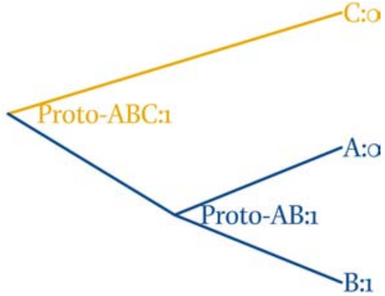


FIGURE 20 Hypothetical tree 1

To calculate the probabilty of the sequence 010 at the tips, we can use the standard tools of probability theory, which come with the following assumptions (Felsenstein 2004: 251):

(4) *Assumptions*
   a. The diachrony of different sites (on a given tree) is independent.
   b. The diachrony of different lineages is independent.

Assumption (4a) means that the probability of the value of one character is not affected by the value of another character. (Our toy example involves only one character, so this assumption will not play a role here, but it is an assumption of the maximum likelihood estimates below in sections 8.5 and 8.6.) Assumption (4b) says that the way in which a character develops in one branch is not dependent on the way in which it develops in another branch.

This latter assumption in particular enables us to calculate the probability of the observed sequence 010 given the above tree as a product of probabilities:

(5) *Calculating likelihood*
$P(X_{\text{Proto-ABC}} = 1) \cdot P(X_{\text{Proto-AB}} = 1 \mid X_{\text{Proto-ABC}} = 1) \cdot P(X_A = 0 \mid X_{\text{Proto-AB}} = 1) \cdot P(X_B = 1 \mid X_{\text{Proto-AB}} = 1) \cdot P(X_C = 0 \mid X_{\text{Proto-ABC}} = 1)$

P( ) is shorthand for "the probability of." So P($X_{\text{Proto-ABC}}$ = 1) refers to the probability that our character X has the value 1 at the root node Proto-ABC. P($X_{\text{Proto-AB}}$ = 1 | $X_{\text{Proto-ABC}}$ = 1) denotes a conditional probability, that is, the probability that our character has the value 1 in Proto-AB given that it was 1 in Proto-ABC. (The information to the right of the pipe '|' is given.) In essence, what we are doing in example (5) is calculating the probability of a series of diachronic events from a particular starting point, namely the value 1 in Proto-ABC.

8.3 *Transition models*

To compute the probabilities in example (5) we need a TRANSITION MODEL. Transition models in biology are probabilistic models of trait evolution (in the context of DNA sequences, they are known as SUBSTITUTION MODELS). The incorporation of such models is a signal feature of maximum likelihood estimation (see further Felsenstein 2004: 156–59, 196–229, Ewens & Grant 2005: 475–95, Nichols & Warnow 2008: 766–69, Baum & Smith 2013: 217–31, Warnow 2018: 146–52). It is possible to encode a range of properties of linguistic change in a model, such as variable base frequencies, variable rates of change, and transition probabilities. Here I focus on two aspects, base frequency and transition probability. For discrete character data such as we have in our dataset, a common trait model is the Mk model (Lewis 2001), which provides formulas for calculating transition probabilities. This is the trait model used below in section 8.5.

The calculation in example (5) requires the probability of the value 1 at the root node. This is known as the BASE FREQUENCY. One way to calculate the probability of each of the character states is to divide by the total possible number of character values. Since there are only two possible values (0 and 1), the probability of each would be 0.5. Alternatively, we could estimate this probability from the relative frequency of the observed data. In our sequence 010, the relative frequency of the value 0 is 2/3, while that of the value 1 is 1/3. In computing the likelihood of the sequence 010 below, I adopt the first method and assign a uniform probability to the values 0 and 1, that is, 0.5.

We need in addition a model that specifies the probability of changing from 0 to 1 and from 1 to 0, as well as the probability of successful transmission from one generation to the next (that is, the probability that the character will not change). In the interest of pedagogical expedience, I stipulate the following transition probabilities:

(6) *Transition probabilities*

a. *The probability of a change of state*
P(1 | 0) = 0.2
P(0 | 1) = 0.2

b. *The probability of no change of state*
P(0 | 0) = 0.8
P(1 | 1) = 0.8

Example (6a) presents two conditional probabilities with a change of state, namely 0 > 1 and 1 > 0. Since language transmission is typically successful and change often slow, I set the probability at 0.2. The probability of no change (i.e., 1 > 1 or 0 > 0) in example (6b) is accordingly 0.8. P(1 | 0) and P(1 | 1) add up to one, as do P(0 | 0) and P(0 | 1), because these two scenarios exhaust the possibility space. That is, given a character value 1, it is certain that it will either change to 0 or remain 1.

8.4 *Computing the likelihood*

Now that we have base frequencies and transition probabilities, we can compute the probabilty of the sequence 010, given the tree above, which I repeat here for convenience:

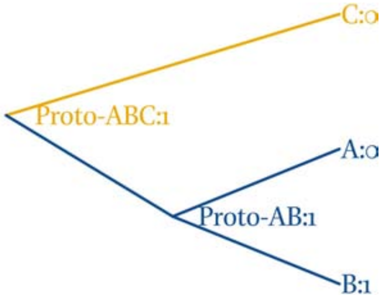


FIGURE 21 Hypothetical tree 1

According to this tree, we would have the following change and non-change events events:

(7) *Change events*
a. Proto-ABC 1 > C 0
b. Proto-AB 1 > A 0

(8) *Non-change events*
a. Proto-ABC 1 > Proto-AB 1
b. Proto-AB 1 > B 1

Given our earlier assumption of independence, the probability can be calculated as the following product:

(9) $P(X_{\text{Proto-ABC}} = 1) \cdot P(X_{\text{Proto-AB}} = 1 \mid X_{\text{Proto-ABC}} = 1) \cdot P(X_A = 0 \mid X_{\text{Proto-AB}} = 1) \cdot P(X_B = 1 \mid X_{\text{Proto-AB}} = 1) \cdot P(X_C = 0 \mid X_{\text{Proto-ABC}} = 1) = 0.5 \cdot 0.8 \cdot 0.2 \cdot 0.8 \cdot 0.2$

This product yields a likelihood of 0.0128 and a log-likelihood of −4.36.

Now that we have computed the likelihood of our character sequence given a particular tree, we can now compare this likelihood score against that of other trees, such as the following, where languages A and C form a clade:

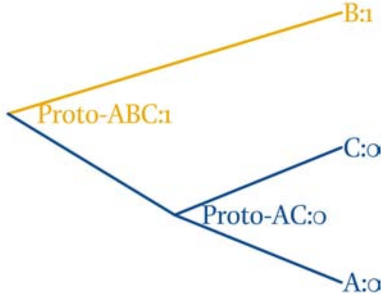


FIGURE 22 Hypothetical tree 2

We observe the same sequence data and now want to know whether this tree is a better fit for that data. To calculate the probability of our data on the above tree, we again calculate the probability of the character value at each point in the tree. According to this tree, we would have the following change and non-change events:

(10) *Change events*
Proto-ABC 1 > Proto-AC 0

(11) *Non-change events*
a. Proto-ABC 1 > B 1
b. Proto-AC 0 > A 0
c. Proto-AC 0 > C 0

We again calculate the product:

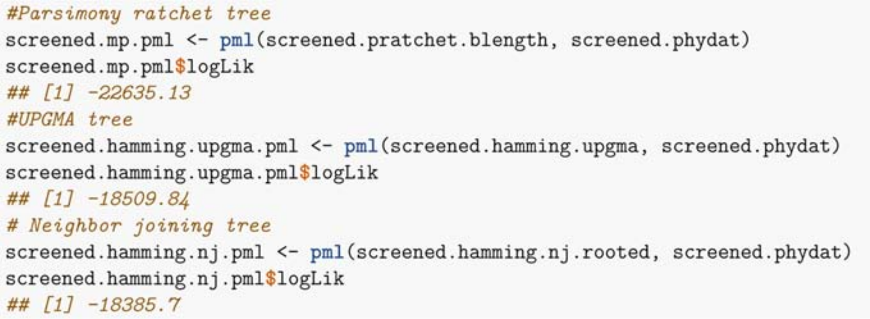
(12) $P(X_{\text{Proto-ABC}} = 1) \cdot P(X_{\text{Proto-AC}} = 0 \mid X_{\text{Proto-ABC}} = 1) \cdot P(X_A = 0 \mid X_{\text{Proto-AC}} = 0) \cdot P(_C = 0 \mid X_{\text{Proto-AC}} = 0) \cdot P(X_B = 1 \mid X_{\text{Proto-ABC}} = 1) = 0.5 \cdot 0.2 \cdot 0.8 \cdot 0.8 \cdot 0.8$

This product yields a likelihood of 0.0512 and a log-likelihood of −2.97. The probability of the character data is therefore higher given hypothetical tree two compared to hypothetical tree one. As the calculations reveal, this is because tree two involves fewer change events that lower the likelihood. Were we to continue trying different topologies, we would eventually identify the tree that makes the data most likely.

This is a highly simplified illustration of maximum likelihood estimation. In practice, maximum likelihood computation is much more complex, for it takes into account factors such as branch length and rate variation among characters. In addition, we usually do not know the ancestral states of characters, so we have to calculate the probability of the observed data across alternative ancestral state scenarios. Nevertheless, the toy example above provides a glimpse of the basic mechanics of the method.

8.5 *Calculate likelihood*

Returning to our Indo-European dataset, we calculate the log-likelihood of the observed data for a given tree with the function `pml()`. The following log-likehood values are based on the Mk model:

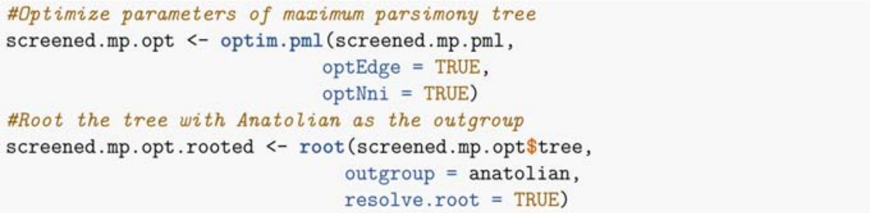


```
#Parsimony ratchet tree
screened.mp.pml <- pml(screened.pratchet.blength, screened.phydat)
screened.mp.pml$logLik
## [1] -22635.13
#UPGMA tree
screened.hamming.upgma.pml <- pml(screened.hamming.upgma, screened.phydat)
screened.hamming.upgma.pml$logLik
## [1] -18509.84
# Neighbor joining tree
screened.hamming.nj.pml <- pml(screened.hamming.nj.rooted, screened.phydat)
screened.hamming.nj.pml$logLik
## [1] -18385.7
```

We have not actually inferred a maximum likelihood tree at this point. We have simply calculated the log-likelihood of the data given three different trees. The lowest log-likelihood score is −18386, which means that the observed data are most likely given the NJ tree inferred in section 7.3 and the Mk model. It is worth emphasizing that these likelihood scores do not refer to the probability of a particular tree. They denote the probability of the *data*, given that tree and its parameters.

8.6 *Maximum likelihood estimation*

The function `pml()` calculates the probability of the data given a particular tree and transition model. It does not identify the tree that maximizes the probability of that data, however. To do that, we use the function `optim.pml()` in `phangorn`, which optimizes model parameters (for more on what can be optimized, type `?optim.pml` into the console).[30] In the following snippet, I optimize two parameters, tree topology and branch length (to conserve space, I only present the code for the maximum parsimony start tree). Note that `optim.pml()` requires an initial `pml` object:[31]



```
#Optimize parameters of maximum parsimony tree
screened.mp.opt <- optim.pml(screened.mp.pml,
                             optEdge = TRUE,
                             optNni = TRUE)
#Root the tree with Anatolian as the outgroup
screened.mp.opt.rooted <- root(screened.mp.opt$tree,
                               outgroup = anatolian,
                               resolve.root = TRUE)
```

The argument `optEdge=TRUE` optimizes branch lengths, while `optNni=TRUE` optimizes the topology. The snippet above adjusts the topology and branch length of the tree until it finds one that yields the highest log-likelihood.[32]

Our MLE trees with optimized topology and branch lengths look as follows (in the interest of space, I do not present the code for the plots for these and subsequent trees):

[30] The packages `diversitree` (FitzJohn 2012) and `CorHMM` (Beaulieu, Oliver, & O'Meara 2017) offer further tools for maximum likelihood phylogenetic inference.

[31] Other maximum likelihood estimation software will carry out a search for the maximum likelihood tree, such as IQ tree (http://iqtree.cibiv.univie.ac.at) or RAxML (Stamatakis 2014).

[32] With larger trees, NNI rearrangements can get stuck in local optima (see section 5.5 above). To circumvent this issue, the argument `rearrangement = "stochastic"` makes stochastic NNI permutations to the tree that then get optimized. Stochastic rearrangement performs a more thorough search for the optimal tree and consequently takes longer to perform.

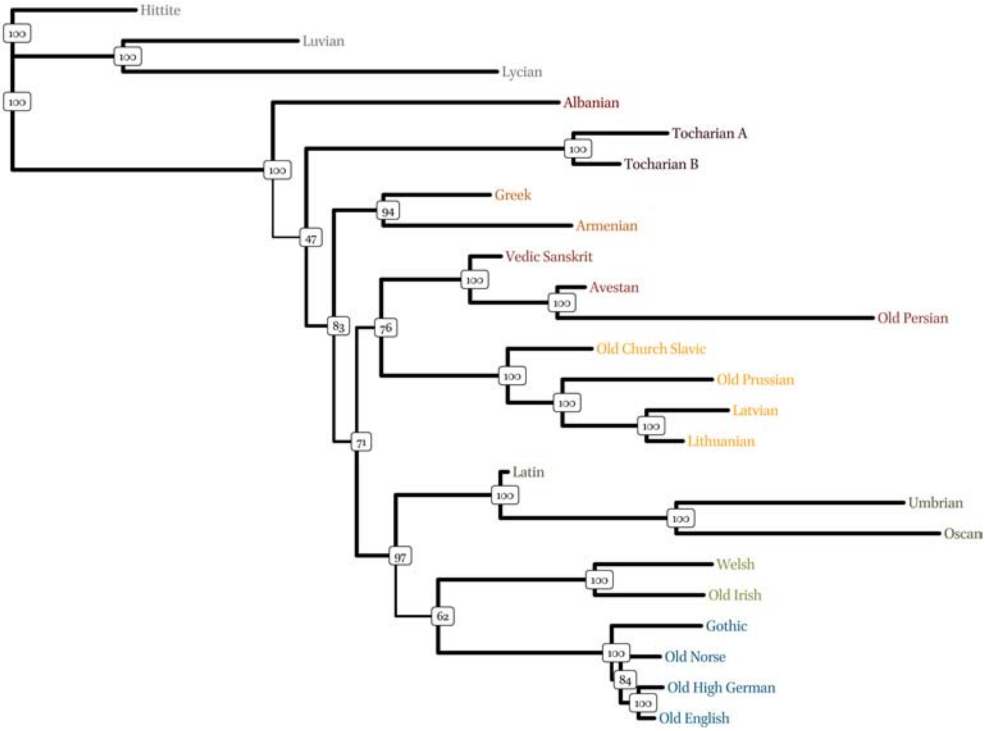


FIGURE 23 MLE tree with bootstrap scores (Ratchet start tree)

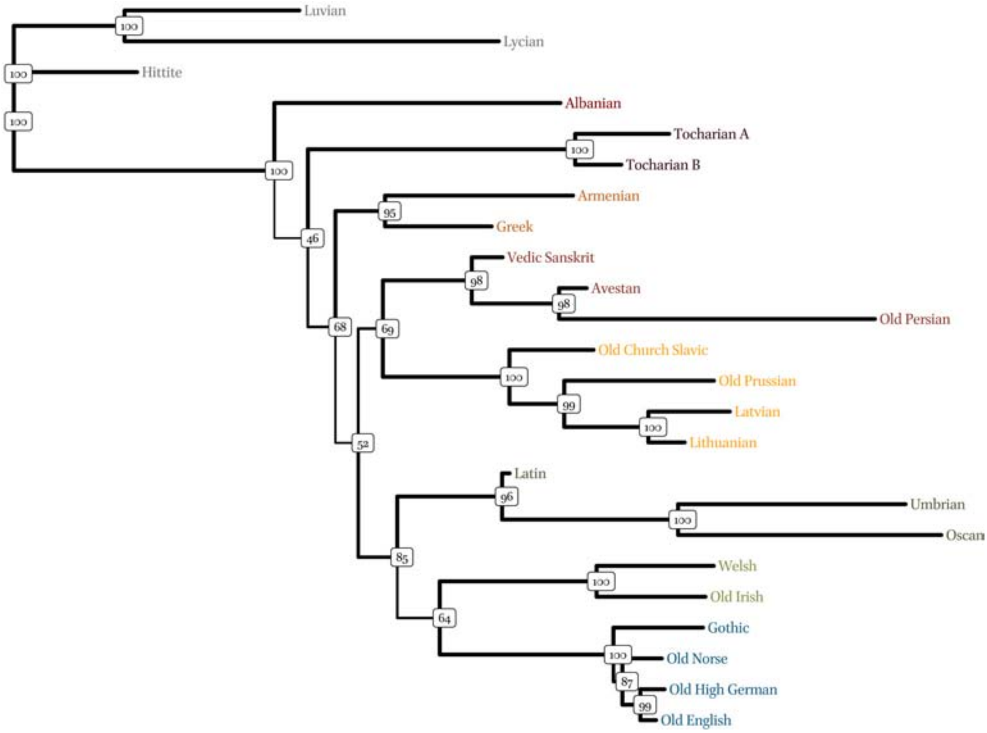


FIGURE 24 MLE tree with bootstrap scores (UPGMA start tree)

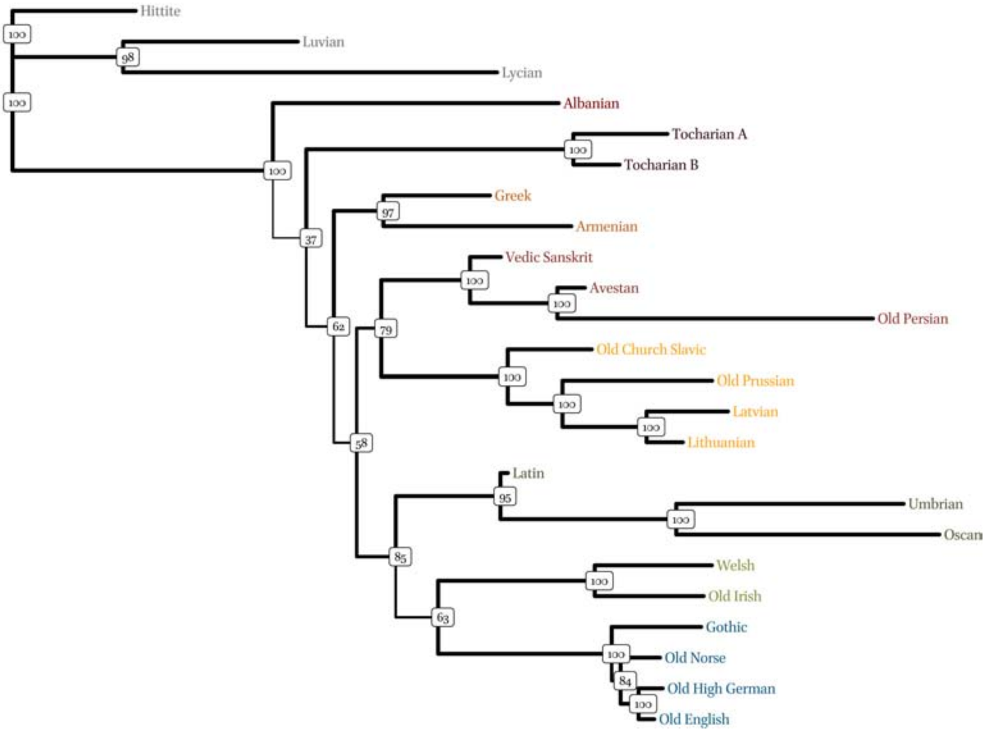


FIGURE 25 MLE tree with bootstrap scores (NJ start tree)

Although we started from three different start trees, maximum likelihood estimation converges on the same topology. The topology of the above trees is close to what we observed with the maximum parsimony trees in section 5 above. It is worth noting that maximum likelihood estimation yields more extreme branch lengths. In all three of the above trees, Oscan, Umbrian, and Old Persian are the most innovative archaic Indo-European languages (as revealed by the length of their branches). One clear flaw with this tree is that Anatolian is presented as almost tantamount to PIE since it is so close to the root.

In inferring the above trees, I did not take full advantage of the capabilities of maximum likehood estimation. For instance, the model used to infer the trees does not contain a parameter for variation in the rates of change. It is well known that rates of linguistic change vary both across and within the components of language, i.e., phonology, morphology, syntax, and the lexicon (e.g., Dixon 1997: 9 n. 1, Nettle 1999b, Gray 2005, Pagel & Meade 2006, Pagel, Atkinson, & Meade 2007, Greenhill et al. 2017). It is possible to incorporate such variation into a model (see, e.g., Yang 2014: 114–20). Phylogenetic estimation based on such a model should accordingly become more accurate given that we are providing the model with more information about the historical processes that give rise to the observable data. There thus remains a lot for historical linguists to explore with likelihood methods.

8.7 *Issues*

One general criticism that has been leveled at maximum likelihood methods is that they do not answer the question that historical linguists are most interested in. Likelihood assesses the probability of the data given a phylogeny and its parameters, but what historical linguists want to know is the probability of a particular phylogeny and a set of parameters given the observed data. Bayesian inference is designed to answer precisely this type of question, since it offers a probability distribution over phylogenetic trees given the observed data (see Drummond & Bouckaert 2015: 19–20 for a comparison between Bayesian and maximum likelihood phylogenetic inference). This is one reason why Bayesian methods have become so prominent in linguistic phylogenetics.

9 Envoi

To sum up, distance-based methods do not perform as well as parsimony or maximum likelihood methods on our dataset. The potential of the latter set of methods remains to be explored. The particulars of the methods aside, it should now be clear that the phylogenies that we infer depend crucially on the assumptions of the method and the data. We need to infer phylogenies from a variety of datasets to determine whether the results obtained are an artifact of that particular dataset or reflect a true phylogenetic signal. This need has long been perceived in the biological sciences:

> Fisher's essential point was that the ability to investigate meaningful population differences from data such as human blood-group frequencies depends on the accumulation of information from a variety of blood-group systems, no one of which will reveal the phylogenetic structure by itself. The same is true today when the wealth of genetic material available for analysis in all species is effectively boundless.
>
> EDWARDS (2009: 6)

For all the promise that advances in computational phylogenetics hold, at the end of the day what matters most is the quality (and quantity) of the input data. On this front, one area that still awaits closer investigation is morphosyntax.[33]

[33] The following remark of Brugmann (1884: 248) remains true today: "Ich zweifle nicht

Bowern (2017: 421), in her review of Pereltsvaig & Lewis (2015), writes that it is a great time to be a historical linguist. She cites the number of new tools that we now have to investigate big questions of language change. I concur, and want to stress that estimation of tree topology is only a small part of what makes the advent of computational phylogenetics so exciting, since these methods enable Indo-Europeanists to pursue questions that were previously out of reach.

The data and code for this tutorial are available at http://doi.org/10.5281/zenodo.3417299.

Acknowledgments

For extremely helpful feedback on earlier versions of the manuscript, I am indebted to Joshua Agee, Arnoud Fournet, Alexei Kassian, Johann-Mattis List, John Lynch, Tyler McCraney, Angelo Mercado, Thomas Olander, Teigo Onishi, Thomas Pellard, Dariusz Piwowarczyk, Ryan Sandell, Matthew Scarborough, Anthony Yates, Dmitrii Zelenskii, and two anonymous reviewers. Klaus Schliep and Felipe Zapata kindly fielded various questions, as well.

References

Agee, Joshua R. 2018. "A glottometric subgrouping of the early Germanic languages." Master's thesis, San José State University.

Albert, Victor, ed. 2005. *Parsimony, phylogeny, and genomics*. Oxford: Oxford University Press.

Aldenderfer, Mark S., & Roger K. Blashfield. 1984. *Cluster analysis*. Newbury Park, CA: Sage.

Andersen, Henning. 2006. "Synchrony, diachrony, & evolution." In *Competing models of linguistic change: Evolution and beyond*, ed. Ole Nedergaard Thomsen, 59–90. Amsterdam: John Benjamins.

Archie, James W., & Joseph Felsenstein. 1993. "The number of evolutionary steps on random and minimum length trees for random evolutionary data." *Theoretical population biology* 43/1: 52–79.

daran, daß tiefer dringende Forschung auf dem Gebiet der vergleichenden Syntax noch manche besondere syntaktische Übereinstimmung zwischen benachbarten Sprachen aufdecken wird." For research on this front, see, e.g., Longobardi & Guardiano (2009), Longobardi et al. (2013), Longobardi et al. (2015), Longobardi & Guardiano (2017).

Atkinson, Quentin D., & Russell D. Gray. 2005. "Curious parallels and curious connections—phylogenetic thinking in biology and historical linguistics." *Systematic biology* 54/4: 513–26.

Babel, Molly, Andrew J. Garrett, Michael Houser, & Maziar Toosarvandani. 2013. "Descent and diffusion in language diversification: A study of Western Numic dialectology." *International journal of American linguistics* 79/4: 445–89.

Barbançon, François, Steven N. Evans, Luay Nakhleh, Donald A. Ringe, & Tandy Warnow. 2013. "An experimental study comparing linguistic phylogenetic linguistic reconstruction methods." *Diachronica* 30/2: 143–70.

Baum, David A., & Stacey D. Smith. 2013. *Tree thinking: An introduction to phylogenetic biology*. Greenwood Village, CO: Roberts and Co.

Baxter, William H. 2006. "Mandarin dialect phylogeny." *Cahiers de linguistique* 35/1: 71–114.

Beaulieu, Jeremy M., Jeffrey C. Oliver, & Brian O'Meara. 2017. *corHMM: Analysis of binary character evolution*. R package version 1.22. Available at https://cran.r-project.org/web/packages/corHMM/index.html.

Bergsten, Johannes. 2005. "A review of long-branch attraction." *Cladistics* 21/2: 163–93.

Boc, Alix, Anna Maria Di Sciullo, & Vladimir Makarenkov. 2010. "Classification of the Indo-European languages using a phylogenetic network approach." In *Classification as a tool for research: Proceedings of the 11th IFCS Biennial Conference and 33rd Annual Conference of the Gesellschaft für Klassifikation e.V., Dresden, March 13–18, 2009*, eds. Hermann Locarek-Junge & Claus Weihs, 647–55. Berlin: Springer.

Borchsenius, Finn, Aymeric Daval-Markussen, & Peter Bakker. 2017. "Phylogenetics in biology and linguistics." In *Creole studies: Phylogenetic approaches*, eds. Peter Bakker, Finn Borchsenius, Carsten Levisen, & Eeva Sippola, 35–58. Amsterdam: John Benjamins.

Bouckaert, Remco R., Philippe Lemey, Michael Dunn, Simon J. Greenhill, Alexander V. Alekseyenko, Alexei J. Drummond, Russell D. Gray, Marc A. Suchard, & Quentin D. Atkinson. 2012. "Mapping the origins and expansion of the Indo-European language family." *Science* 337/6097: 957–60.

Bouckaert, Remco R., Timothy G. Vaughan, Joëlle Barido-Sottani, Sebastián Duchêne, Mathieu Fourment, Alexandra Gavryushkina, Joseph Heled, et al. 2019. "BEAST 2.5: An advanced software platform for Bayesian evolutionary analysis." *PLoS computational biology* 15/4: e1006650.

Bowern, Claire. 2018. "Computational phylogenetics." *Annual review of linguistics* 4: 281–96.

Bowern, Claire. 2017. "The Indo-European controversy and Bayesian phylogenetic methods." *Diachronica* 34/3: 421–36.

Bowern, Claire, & Harold J. Koch. 2004. "Subgrouping methodology in historical linguistics." In *Australian languages: Classification and the comparative method*, eds. Claire Bowern & Harold J. Koch, 1–15. Amsterdam: John Benjamins.

Brugmann, Karl. 1884. "Zur Frage nach den Verwandtschaftsverhältnissen der indogermanischen Sprachen." *Internationale Zeitschrift für allgemeine Sprachwissenschaft* 1: 228–56.

Chang, Will, Chundra Cathcart, David P. Hall, & Andrew J. Garrett. 2015. "Ancestry-constrained phylogenetic analysis supports the Indo-European steppe hypothesis." *Language* 91/1: 194–244.

Clackson, James P.T. 2000. "Time depth in Indo-European." In *Time depth in historical linguistics: Papers in the prehistory of languages*, eds. Colin A. Renfrew, April M.S. McMahon, & Larry Trask, 441–54. Cambridge: McDonald Institute for Archaeological Research.

Croft, William A. 2008. "Evolutionary linguistics." *Annual review of anthropology* 37/1: 219–34.

Darwin, Charles. 1882. *The descent of man, and selection in relation to sex*. London: John Murray.

DeLisi, Jessica. 2018. "Armenian prosody in typology and diachrony." *Language dynamics and change* 8/1: 108–33.

Delmestri, Antonella, & Nello Cristianini. 2010. "Linguistic phylogenetic inference by PAM-like matrices." Tech Report DISI-10-058. University of Trento.

Dixon, Robert M.W. 1997. *The rise and fall of languages*. Cambridge: Cambridge University Press.

Drinka, Bridget. 2013. "Phylogenetic and areal models of Indo-European relatedness: The role of contact in reconstruction." *Journal of language contact* 6/2: 379–410.

Drummond, Alexei J., & Remco R. Bouckaert. 2015. *Bayesian evolutionary analysis with BEAST*. Cambridge: Cambridge University Press.

Dunn, Michael. 2015. "Language phylogenies." In *The Routledge handbook of historical linguistics*, eds. Claire Bowern & Bethwyn Evans, 190–211. London: Routledge.

Durbin, Richard, Sean R. Eddy, Anders Krogh, & Graeme Mithcison. 1998. *Biological sequence analysis: Probabilistic models of proteins and nucleic acids*. Cambridge: Cambridge University Press.

Eckardt, Regine, Gerhard Jäger, & Tonjes Veenstra, eds. 2008. *Variation, selection, development: Probing the evolutionary model of language change*. Berlin: de Gruyter.

Edwards, Anthony W.F. 2009. "Statistical methods for evolutionary trees." *Genetics* 183/1: 5–12.

Efron, Bradley. 1979. "Bootstrap methods: Another look at the jackknife." *The annals of statistics* 7/1: 1–26.

Efron, Bradley. 2003. "Second thoughts on the bootstrap." *Statistical science* 18/2: 135–40.

Efron, Bradley, Elizabeth Halloran, & Susan Holmes. 1996. "Bootstrap confidence levels for phylogenetic trees." *Proceedings of the National Academy of Sciences* 93/23: 13429–34.

Efron, Bradley, & Robert Tibshirani. 1993. *An introduction to the bootstrap*. New York: Chapman; Hall.

Egan, Mary G. 2006. "Support versus corroboration." *Journal of biomedical informatics* 39/1: 72–85.

Enfield, Nick J. 2014. *Natural causes of language: Frames, biases, and cultural transmission*. Berlin: Language Science Press.

Everitt, Brian S., Sabine Landau, Morven Leese, & Daniel Stahl. 2011. *Cluster analysis*. 5th ed. Chichester: Wiley.

Ewens, Warren, & Gregory Grant. 2005. *Statisical methods in bioinformatics: An introduction*. 2nd ed. New York: Springer.

Farris, James S. 1977. "Phylogenetic analysis under Dollo's Law." *Systematic zoology* 26/1: 77–88.

Farris, James S. 1989. "The retention index and rescaled consistency index." *Cladistics* 5/4: 417–19.

Felsenstein, Joseph. 2004. *Inferring phylogenies*. Oxford: Oxford University Press.

Felsenstein, Joseph. 1978a. "The number of evolutionary trees." *Systematic zoology* 27/1: 27–33.

Felsenstein, Joseph. 1978b. "Cases in which parsimony or compatability methods will be positively misleading." *Systematic zoology* 27/4: 401–10.

Felsenstein, Joseph. 1981. "Evolutionary trees from DNA sequences: A maxumum likelihood approach." *Journal of molecular evolution* 17/6: 368–76.

Felsenstein, Joseph. 1985. "Confidence limits on phylogenies: An approach using the bootstrap." *Evolution* 39/4: 783–91.

Felsenstein, Joseph. n.d. "Phylogeny programs." http://evolution.gs.washington.edu/phylip/software.html#Parsimony.

Fitch, Walter M. 1971. "Toward defining the course of evolution: Minimum change for a specified tree topology." *Systematic zoology* 20/4: 406–16.

FitzJohn, Richard G. 2012. "Diversitree: Comparative phylogenetic analyses of diversification in R." *Methods in ecology and evolution* 3/6: 1084–92.

François, Alexandre. 2015. "Trees, waves and linkages: Models of language diversification." In *The Routledge handbook of historical linguistics*, eds. Claire Bowern & Bethwyn Evans, 161–89. London: Routledge.

Garde, Paul. 1961. "Réflexions sur les différences phonétiques entre les langues slaves." *Word* 17/1: 34–62.

Garrett, Andrew J. 1999. "A new model of Indo-European subgrouping and dispersal." In *Proceedings of the Twenty-Fifth Annual Meeting of the Berkeley Linguistics Society, February 12–15, 1999*, eds. Steve S. Chang, Lily Liaw, & Josef Ruppenhofer, 146–56. Berkeley: Berkeley Linguistics Society.

Garrett, Andrew J. 2006. “Convergence in the formation of Indo-European subgroups: Phylogeny and chronology.” In *Phylogenetic methods and the prehistory of languages*, eds. Peter Forster & Colin A. Renfrew, 139–51. Cambridge: McDonald Institute for Archaeological Research.

Garrett, Andrew J. 2018. “New perspectives on Indo-European phylogeny and chronology.” *Proceedings of the American Philosophical Society* 162/1: 25–38.

Garrett, Andrew J., & Keith Johnson. 2013. “Phonetic bias in sound change.” In *Origins of sound change*, ed. Alan C.L. Yu, 51–97. Oxford: Oxford University Press.

Gascuel, Olivier, ed. 2007. *Mathematics of evolution and phylogeny*. Oxford: Oxford University Press.

Geisler, Hans, & Johann-Mattis List. 2010. “Beautiful trees on unstable ground: Notes on the data problem in lexicostatistics.” https://hal.archives-ouvertes.fr/hal-01298493.

Graur, Dan, & Wen-Hsiung Li. 2000. *Fundamentals of molecular evolution*. 2nd ed. Sunderland, MA: Sinauer.

Gray, Russell D. 2005. “Evolution: Pushing the time barrier in the quest for language roots.” *Science* 309/5743: 2007–08.

Gray, Russell D., & Quentin D. Atkinson. 2003. “Language-tree divergence times support the Anatolian theory of Indo-European origin.” *Nature* 426/6965: 435–39.

Greenhill, Simon J., Chieh-Hsi Wu, Xia Hua, Michael Dunn, & Stephen C. Levinson. 2017. “Evolutionary dynamics of language systems.” *Proceedings of the National Academy of Sciences of the United States of America* 114/42: E8822–29.

Gries, Stefan Th. 2013. *Statistics for linguistics with R: A practical introduction*. Berlin: de Gruyter.

Gries, Stefan Th. 2017. *Quantitative corpus linguistics with R: A practical introduction*. 2nd ed. London: Routledge.

Grolemund, Garrett, & Hadley Wickham. 2017. *R for data science: Import, tidy, transform, visualize, and model data*. Sebastopol: O’Reilly.

Hall, Barry G. 2018. *Phylogenetic trees made easy: A how-to manual*. 5th ed. Sunderland, MA: Sinauer.

Hamilton, Andrew. 2013. *The evolution of phylogenetic systematics*. Berkeley: University of California Press.

Hauser, David L., & George Boyajian. 1997. “Proportional change and patterns of homoplasy: Sanderson and Donoghue revisited.” *Cladistics* 13/1–2: 97–100.

Heggarty, Paul. 2006. “Interdisciplinary indiscipline?” In *Phylogenetic methods and the prehistory of languages*, eds. Peter Forster & Colin A. Renfrew, 183–94. Cambridge: McDonald Institute for Archaeological Research.

Hoenigswald, Henry M. 1960. *Language change and linguistic reconstruction*. Chicago: University of Chicago Press.

Hoenigswald, Henry M., & Linda F. Weiner, eds. 1987. *Biological metaphor and cladistic classification: An Interdisciplinary Approach*. London: Frances Pinter.

Höhna, Sebastian, Michael J. Landis, Tracy A. Heath, Bastien Boussau, Nicolas Lartillot, Brian R. Moore, John P. Huelsenbeck, & Fredrik Ronquist. 2016. "RevBayes: Bayesian phylogenetic inference using graphical models and an interactive model-specification language." *Systematic biology* 65/4: 726–36.

Holden, Clare Janaki. 2002. "Bantu language trees reflect the spread of farming across sub-Saharan Africa: Biological sciences." *Proceedings of the Royal Society B* 269: 793–99.

Huelsenbeck, John P., & Keith A. Crandall. 1997. "Phylogeny estimation and hypothesis testing using maximum likelihood." *Annual review of ecology and systematics* 28/1: 437–66.

Huson, Daniel H., Regula Rupp, & Celine Scornavacca. 2010. *Phylogenetic networks: Concepts, algorithms and applications*. Cambridge: Cambridge University Press.

Jacques, Guillaume, & Johann-Mattis List. 2018. "Save the trees: Why we need tree models in linguistic reconstruction (and when we should apply them)." *Journal of historical linguistics* 9/1: 128–66.

Jäger, Gerhard. 2015. "Support for linguistic macrofamilies from weighted sequence alignment." *Proceedings of the National Academy of Sciences of the United States of America PNAS, Proceedings of the National Academy of Sciences* 112/41: 12752–57.

Johnson, Keith. 2008. *Quantitative methods In linguistics*. Malden, MA: Blackwell.

Kassambara, Alboukadel. 2017. *Practical guide to cluster analysis in R: Unsupervised machine learning*. STHDA.

Kessler, Brett. 2001. *The significance of word lists: Statistical tests for investigating historical connections between languages*. Stanford: Center for the Study of Language and Information.

Kitching, Ian J., Peter L. Forey, Christopher J. Humphries, & David M. Williams. 1998. *Cladistics: The theory and practice of parsimony analysis*. Oxford: Oxford University Press.

Klingenberg, Christian Peter, & Nelly A. Gidaszewski. 2010. "Testing and quantifying phylogenetic signals and homoplasy in morphometric data." *Systematic biology* 59/3: 245–61.

Lass, Roger. 1997. *Historical linguistics and language change*. Cambridge: Cambridge University Press.

Levshina, Natalia. 2015. *How to do linguistics with R*. Amsterdam: John Benjamins.

Lewis, Martin L., & Asya Pereltsvaig. 2012. "Linguistic phylogenies are not the same as biological phylogenies." http://www.geocurrents.info/cultural-geography/linguistic-geography/linguistic-phylogenies-are-not-the-same-as-biological-phylogenies#ixzz5XN2YwEcV.

Lewis, Paul O. 1998. "Maximum likelihood as an alternative to parsimony for inferring phylogeny using nucleotide sequence data." In *Molecular systematics of plants*, eds. Douglas E. Soltis, Pamela S. Soltis, & Jeff J. Doyle, vol. 2, 132–63. Boston: Springer.

Lewis, Paul O. 2001. "A likelihood approach to estimating phylogeny from discrete morphological character data." *Systematic biology* 50/6: 913–25.

Lipscomb, Diana Leigh. 1998. "Basics of cladistic analysis." https://www2.gwu.edu/~clade/faculty/lipscomb/Cladistics.pdf.

List, Johann-Mattis. 2017. "Historical language comparison with LingPy and EDICTOR." Jena: Max Planck Institute for the Science of Human History, Linguistic and Cultural Evolution. https://github.com/digling/edictor-tutorial.

Longobardi, Giuseppe, & Cristina Guardiano. 2009. "Evidence for syntax as a signal of historical relatedness." *Lingua* 119/11: 1679–1706.

Longobardi, Giuseppe, & Cristina Guardiano. 2017. "Phylogenetic reconstruction in syntax: The parametric comparison method." In *The Cambridge handbook of historical syntax*, eds. Adam Ledgeway & Ian G. Roberts, 241–72. Cambridge: Cambridge University Press.

Longobardi, Giuseppe, Cristina Guardiano, Giuseppina Silvestri, Alessio Boattini, & Andrea Ceolin. 2013. "Toward a syntactic phylogeny of modern Indo-European languages." *Journal of historical linguistics* 3/1: 122–52.

Longobardi, Giuseppe, Cristina Guardiano, Giuseppina Silvestri, Alessio Boattini, & Andrea Ceolin. 2015. "Toward a syntactic phylogeny of modern Indo-European languages." In *Proto-Indo-European syntax and its development*, eds. Leonid I. Kulikov & Nikolaos Lavidas, 125–56. Amsterdam: John Benjamins.

Mallory, James P., & Douglas Q. Adams. 2006. *The Oxford introduction to Proto-Indo-European and the Proto-Indo-European world*. Oxford: Oxford University Press.

Maurits, Luke, Robert Forkel, Gereon A. Kaiping, & Quentin D. Atkinson. 2017. "BEASTling: A software tool for linguistic phylogenetics using BEAST 2." *PLoS ONE* 12/8: 1–17.

McMahon, April M.S., & Robert McMahon. 2005. *Language classification by numbers*. Oxford: Oxford University Press.

Melchert, H. Craig. Forthcoming. "The position of Anatolian." In *The Oxford handbook of Indo-European studies*, eds. Michael Weiss & Andrew J. Garrett. Oxford: Oxford University Press.

Melchert, H. Craig, & Norbert Oettinger. 2009. "Ablativ und Instrumental im Hethitischen und Indogermanischen: Ein Beitrag zur relativen Chronologie." *Incontri linguistici* 32: 53–73.

Morrison, David A. 2011. *Introduction to phylogenetic networks: Introduction to phylogenetic networks*. Uppsala: RJR Productions.

Mounce, Ross. 2013. "Comparative cladistics: Fossils, morphological data partitions and lost branches in the fossil tree of life." PhD thesis, University of Bath.

Nakhleh, Luay, Donald A. Ringe, & Tandy Warnow. 2005. "Perfect phylogenetic networks: A new methodology for reconstructing the evolutionary history of natural languages." *Language* 81/2: 382–420.

Nakhleh, Luay, Tandy Warnow, Donald A. Ringe, & Steven N. Evans. 2005. "A comparison of phylogenetic linguistic reconstruction methods on an Indo-European dataset." *Transactions of the Philological Society* 103/2: 171–92.

Nettle, Daniel. 1999a. *Linguistic diversity*. Oxford: Oxford University Press.

Nettle, Daniel. 1999b. "Is the rate of linguistic change constant?" *Lingua* 108/2–3: 119–36.

Nichols, Johanna, & Tandy Warnow. 2008. "Tutorial on computational linguistic phylogeny." *Language and linguistics compass* 2/5: 760–820.

Nixon, Kevin C. 1999. "The parsimony ratchet, a new method for rapid parsimony analysis." *Cladistics* 15/4: 407–14.

Nunn, Charles L. 2011. *The comparative approach in evolutionary anthropology and biology*. Chicago: University of Chicago Press.

Oaks, Jamie R. 2015. "Bayesian phylogenetics: Methods, algorithms, and applications.—Edited by Ming-Hui Chen, Lynn Kuo, & Paul O. Lewis." *Systematic biology* 64/6: 1122–25.

Olander, Thomas. 2018. "Connecting the dots: The Indo-European family tree as a heuristic device." In *Proceedings of the 29th Annual UCLA Indo-European Conference*, eds. David M. Goldstein, Stephanie W. Jamison, & Brent Vine, 181–202. Bremen: Hempen.

Pagel, Mark. 2000. "Maximum-likelihood models for glottochronology and for reconstructing linguistic phylogenies." In *Time depth in historical linguistics*, eds. Colin A. Renfrew, April M.S. McMahon, & R. Larry Trask, vol. 1, 189–207. Cambridge: McDonald Institute for Archaeological Research.

Pagel, Mark. 2009. "Human language as a culturally transmitted replicator." *Nature Reviews Genetics* 10/6: 405–15.

Pagel, Mark. 2017. "Darwinian perspectives on the evolution of human languages." *Psychonomic bulletin & review* 24/1: 151–57.

Pagel, Mark, Quentin D. Atkinson, & Andrew Meade. 2007. "Frequency of word-use predicts rates of lexical evolution throughout Indo-European history." *Nature* 449/7163: 717–20.

Pagel, Mark, & Andrew Meade. 2006. "Estimating rates of lexical replacement on phylogenetic trees of languages." In *Phylogenetic methods and the prehistory of languages*, eds. Peter Forster & Colin A. Renfrew, 173–82. Cambridge: McDonald Institute for Archaeological Research.

Paradis, Emmanuel. 2012. *Analysis of phylogenetics and evolution with R*. 2nd ed. Dordrecht: Springer.

Pawitan, Yudi. 2001. *In all likelihood: Statistical modelling and likelihood using inference*. Oxford: Oxford University Press.

Penzl, Herbert. 1960. "Hoenigswald on linguistic change and linguistic reconstruction." *American speech* 35: 216–19.

Pereltsvaig, Asya, & Martin L. Lewis. 2015. *The Indo-European controversy: Facts and fallacies in historical linguistics*. Cambridge: Cambridge University Press.

R Core Team. 2019. *R: A language and environment for statistical computing*. Vienna: R Foundation for Statistical Computing.

Revell, Liam J., & Scott A. Chamberlain. 2015. "Package `Rphylip`." https://cran.r-project.org/web/packages/Rphylip/Rphylip.pdf.

Rexová, Kateřina, Daniel Frynta, & Jan Zrzavý. 2003. "Cladistic analysis of languages: Indo-European classification based on lexicostatistical data." *Cladistics* 19/2: 120–27.

Ringe, Donald A. 2017. "Indo-European dialectology." In *Comparative Indo-European linguistics: An international handbook of language comparison and the linguistic reconstruction of Indo-European*, eds. Jared S. Klein, Brian D. Joseph, & Matthias Fritz, vol. 1, 62–75. Berlin: de Gruyter.

Ringe, Donald A., & Ann Taylor. 2002. "The Indo-European word lists." https://www.cs.rice.edu/~nakhleh/CPHL/ie-wordlist-07.pdf.

Ringe, Donald A., & Ann Taylor. 2007a. "Morphological characters." https://www.cs.rice.edu/~nakhleh/CPHL/code-m-07.pdf.

Ringe, Donald A., & Ann Taylor. 2007b. "Phonological characters." https://www.cs.rice.edu/~nakhleh/CPHL/code-p-07.pdf.

Ringe, Donald A., Tandy Warnow, & Ann Taylor. 2002. "Indo-European and computational cladistics." *Transactions of the Philological Society* 100/1: 59–129.

Ringe, Donald A., Tandy Warnow, & Ann Taylor. 2012. "Cognations of lexical characters." https://www.cs.utexas.edu/users/tandy/cognations-2k-revised.pdf.

Rosenbach, Anette. 2008. "Language change as cultural evolution: Evolutionary approaches to language change." In *Variation, selection, development: Probing the evolutionary model of language change*, eds. Regine Eckardt, Gerhard Jäger, & Tonjes Veenstra, 23–72. Berlin: de Gruyter.

Saitou, Naruya, & Masatoshi Nei. 1987. "The neighbor-joining method: A new method for reconstructing phylogenetic trees." *Molecular biology and evolution* 4/4: 406–25.

Sanderson, Michael J. 1989. "Confidence limits on phylogenies: The bootstrap revisited." *Cladistics* 5/2: 113–29.

Sanderson, Michael J. 1995. "Objections to bootstrapping phylogenies: A critique." *Systematic biology* 44/3: 299–320.

Sanderson, Michael J., & Michael J. Donoghue. 1989. "Patterns of variations in levels of homoplasy." *Evolution* 43/8: 1781–95.

Sankoff, David. 1975. "Minimal mutation trees of sequences." *SIAM journal of applied mathematics* 28/1: 35–42.

Scarborough, Matthew J.C. 2016. "The Aeolic dialects of ancient Greek: A study in historical dialectology and linguistic classification." PhD thesis, University of Cambridge.

Schleicher, August. 1863. *Die Darwinsche Theorie und die Sprachwissenschaft—offenes Sendschreiben an Herrn Dr. Ernst Haeckel*. Weimar: Böhlau.

Schliep, Klaus P. 2011. "`phangorn`: phylogenetic analysis in R." *Bioinformatics* 27/4: 592–93.

Schliep, Klaus P. 2017. "Using binary or discrete data with `phangorn`." https://kschliep.netlify.com/post/binary2phydat/.

Schliep, Klaus P. 2018a. "Splits and networkx." https://cran.r-project.org/web/packages/phangorn/vignettes/Networx.html.

Schliep, Klaus P. 2018b. "Estimating phylogenetic trees with `phangorn`." https://cran.r-project.org/web/packages/phangorn/vignettes/Trees.pdf.

Schmidt, Heiko A., & Arndt von Haeseler. 2009. "Phylogenetic inference using maximum likelihood methods." In *The phylogenetic handbook*, eds. Philippe Lemey, Marco Salemi, & Anne-Mieke Vandamme, 181–209. Cambridge: Cambridge University Press.

Schmidt, Johannes. 1872. *Die verwantschaftsverhältnisse der indogermanischen sprachen*. Weimar: H. Böhlau.

Schuchardt, Hugo. 1900. *Über die Klassifikation der romanischen Mundarten: Probe-Vorlesung, gehalten zu Leipzig am 30. April 1870*. Graz: Kaiserlich-königliche Universitäts-Buchdruckerei Graz.

Schulmeister, Susanne. 2004. "Inconsistency of maximum parsimony revisited." *Systematic biology* 53/4: 521–28.

Semple, Charles, & Mike Steel. 2003. *Phylogenetics*. Oxford: Oxford University Press.

Silverman, Daniel Doron. 2012. *Neutralization*. Cambridge: Cambridge University Press.

Skelton, Christina. 2015. "Borrowing, character weighting, and preliminary cluster analysis in a phylogenetic analysis of the ancient Greek dialects." *Indo-European linguistics* 3/1: 84–117.

Sneath, Peter H.A., & Robert R. Sokal, eds. 1973. *Numerical taxonomy*. San Francisco: W.H. Freeman.

Sober, Elliott. 1988. *Reconstructing the past: Parsimony, evolution, and inference*. Oxford: Oxford University Press.

Sober, Elliott. 2015. *Ockham's razors: A user's manual*. Cambridge: Cambridge University Press.

Sokal, Robert R., & Charles D. Michener. 1958. "A statistical method for evaluating systematic relationships." *University of Kansas scientific bulletin* 28/22: 1409–38.

Sokal, Robert R., & James R. Rohlf. 1994. *Biometry: The principles and practices of statistics in biological research*. 4th ed. New York: W.H. Freeman.

Stamatakis, Alexandros. 2014. "RAxML version 8: A tool for phylogenetic analysis and post-analysis of large phylogenies." *Bioinformatics* 30/9: 1312–13.

Steel, Mike. 2016. *Phylogeny: Discrete and random processes in evolution*. Philadelphia: Society for Industrial and Applied Mathematics.

Stewart, Caro-Beth. 1993. "The powers and pitfalls of parsimony." *Nature* 361/6413: 603–07.

Swofford, David L., & Wayne P. Maddison. 1987. "Reconstructing ancestral states under Wagner parsimony." *Mathematical biosciences* 87/2: 199–229.

Swofford, David L., Gary J. Olsen, Peter J. Waddell, & David M. Hillis. 1996. "Phylogenetic inference." In *Molecular systematics*, eds. David M. Hillis, Craig Moritz, & Barbara K. Mable, 2nd ed., 407–514. Sunderland, MA: Sinauer.

Swofford, David L., & Jack Sullivan. 2009. "Phylogeny inference based on parsimony and other methods using PAUP*." In *The phylogenetic handbook*, eds. Philippe Lemey, Marco Salemi, & Anne-Mieke Vandamme, 267–312. Cambridge: Cambridge University Press.

Taylor, Ann, Tandy Warnow, & Donald A. Ringe. 2000. "Character-based linguistic reconstruction of a cladogram." In *Historical linguistics 1995*, eds. John Charles Smith & Delia Bentley, vol. 1, 393–408. Amsterdam: John Benjamins.

Van de Peer, Yves. 2009. "Phylogenetic inference based on distance methods." In *The phylogenetic handbook*, eds. Philippe Lemey, Marco Salemi, & Anne-Mieke Vandamme, 142–60. Cambridge: Cambridge University Press.

Verkerk, Annemarie. 2017. "Phylogenies: Future, not fallacy." *Language dynamics and change* 7/1: 127–40.

Warnow, Tandy. 2018. *Computational phylogenetics: An introduction to designing methods for phylogeny estimation*. Cambridge: Cambridge University Press.

Wenzel, John. 2002. "Phylogenetic analysis: The basic method." In *Techniques in molecular systematics and evolution*, eds. Rob DeSalle, Gonzalo Giribet, & Ward Wheeler, 4–30. Basel: Birkhäuser.

Wichmann, Søren, & Arpiar Saunders. 2007. "How to use typological databases in historical linguistic research." *Diachronica* 24/2: 373–404.

Wickham, Hadley. 2014. *Advanced R*. Boca Raton: CRC Press.

Widmer, Paul. 2018. "Indogermanische Stammbäume: Datentypen, Methoden." In *100 Jahre Entzifferung des Hethitischen: Morphosyntaktische Kategorien in Sprachgeschichte und Forschung. Akten der Arbeitstagung der Indogermanischen Gesellschaft vom 21. bis 23. September 2015 in Marburg*, eds. Elisabeth Rieken, Ulrich Geupel, & Theresa Maria Roth, 373–88. Wiesbaden: Reichert.

Wiens, John J. 2000. *Phylogenetic analysis of morphological data*. Washington, D.C.: Smithsonian Books.

Wiley, Edward O., & Bruce S. Lieberman. 2011. *Phylogenetics: Theory and practice of phylogenetic systematics*. 2nd ed. Hoboken: Wiley-Blackwell.

Yang, Ziheng. 2014. *Molecular evolution: A statistical approach*. Oxford: Oxford University Press.

Yu, Guangchuang, David Smith, Huachen Zhu, Yi Guan, & Tommy Tsan-Yuk Lam. 2017. "`ggtree`: An R package for visualization and annotation of phylogenetic trees with their covariates and other associated data." *Methods in ecology and evolution* 8/1: 28–36.