Universität Leipzig

Fakultät für Mathematik und Informatik
Institut für Informatik

Phylogenetic Targeting: A Systematic Approach and Computer Program for Targeting Research Effort in Comparative Evolutionary Biology

Diplomarbeit

Aufgabenstellung und Betreuung:
Dr. Charles Nunn
Prof. Peter F. Stadler

vorgelegt von:
Christian Arnold

Studiengang:
Informatik

Studienrichtung:
Bioinformatik

Leipzig, Juni 2008

Abstract

For centuries, findings from comparative studies have provided new insights into the process of evolution. For many of the most interesting questions in biology a variety of hypotheses have been proposed, but exceptionally few species have been studied to test these hypotheses comparatively. But which species should be studied? By taking the hypotheses, comparative data relevant to the question of interest and a specific phylogeny, we can identify which species would provide the most compelling tests of hypotheses. This is particularly important, because studying species is both expensive and time-consuming. It is achieved by generating all possible pairwise comparisons and scoring them in relation to their relevance, and can also be used to identify missing data points in the phylogeny. Using this method of 'phylogenetic targeting', gaps in our knowledge of particular lineages as well as in relation to particular biological traits of interest can be revealed. These gaps, if systematically biased towards particular species or lineages, can generate noticeable statistical biases in comparative studies. Additionally, we developed a web-based, freely available and publicly accessible computer program, *PhyloTargeting*, which implements this idea. In summary, we provide a new systematic, quantitative and phylogenetic approach to identifying where future research effort should be placed. As an example, we apply a sleep dataset to the program to identify key species that need to be studied to test hypotheses related to the function and evolution of sleep, such as if sleep is profitable for the brain or not.

Zusammenfassung

Seit Jahrzehnten haben Vergleichsstudien neue Erkenntisse über den Evolutionsprozess geliefert. Für viele wichtige biologische Fragestellungen wurden eine Reihe von Hypothesen vorgeschlagen, aber außergewöhnlich wenige Spezies wurden studiert, um diese Hypothesen vergleichend zu testen. Doch welche Spezies sollten studiert werden? Gibt es phylogenetische Verzerrungen bei schon vorhandenen Daten? Durch Generierung aller möglichen paarweisen Kombinationen von Spezies und anschließender Bewertung mittels geeigneter Scoringfunktionen können wir Spezies identifizieren, die am aussagekräftigsten für zukünftige Datenerhebung sind. Dies ist besonders wichtig, denn Datenerhebung ist oft zeitaufwändig und teuer. Ein weiteres Anwendungsgebiet ist die Aufdeckung von fehlenden Datenpunkten in der Phylogenie. Benötigt werden dafür nur konkrete Hypothesen, Daten relevant zur Fragestellung, und eine konkreten Phylogenie. Durch "phylogenetisches Targeting" können also Wissenslücken in Bezug auf Abstammungen oder biologische Fragestellungen offenbart werden. Diese Lücken können, falls systematisch verzerrt, einen deutlichen Bias in Vergleichsstudien erzeugen. Zusätzlich haben wir ein web-basiertes, frei verfügbares und öffentlich zugängiges Computerprogramm, *PhyloTargeting*, entwickelt, dass diese Idee umsetzt. Zusammenfassend stellen wir also einen neuen, sytematischen, quantitativen und phylogenetischen Ansatz vor, um die Richtung von zukünftigen Forschungsaufwand zu konkretisieren. Als Beispiel wird ein Datensatz mit Hypothesen zur Evolution und Funktion von Schlaf mithilfe des Programmes ausgewertet. Insbesondere identifizieren wir Schlüsselspezies für die Hypothese, dass Schlaf für das Gehirn profitabel ist.

Acknowledgements

I wish to thank all involved people who contributed to the success of this work. First and foremost I would like to thank Charles Nunn, who was responsible for both idea and specific implementation ideas for the developed application. His expertise and optimism also kept my motivation alive, and it was always fun working with him. Furthermore, Peter Stadler contributed significantly to the usefulness and applicability of the program. He always set an impetus for the success of this work, and his analytical skills and abilities to abstract complex processes and algorithms impressed me greatly. I also want to thank all of my friends, who have patiently waited while I completed my thesis. Especially my girlfriend Nicole, who gave me unconditional support at all times. She gave me hope when I needed it, despite the plethora of work that kept me constantly busy. I thank all proofreaders (Kyle Barbour, Erin Leigh Flynn, Tasmina Tazeen, and Olaf Thalmann) for correcting errors and bad style. I hope I can return the favor someday! A last, but very grateful thanks is for my family. They supported me selfishly over all these years, they gave me optimism and courage, and without them, all this work would not have been possible.

Danksagung

Ich möchte mich bei allen Leuten, die zum Erfolg dieser Arbeit beigetragen haben, herzlich bedanken. Allen voran Charles Nunn, der sowohl für die Idee des Projektes als auch für konkrete Umsetzungen verantwortlich war. Sein Fachwissen und sein Optimismus haben die Motivation für das Projekt stets am Leben erhalten, und es hat stets großen Spaß gemacht, mit ihm zusammen zu arbeiten. Zudem hat Peter Stadler erheblich zum Nutzen und zur Anwendung des Programms beigetragen. Auch sonst hat er stets positive Impulse für das Gelingen gesetzt, und seine analytischen Fähigkeiten sowie sein Abstraktionsvermögen von komplexen Zusammenhängen haben mich sehr beeindruckt. Mein Dank gilt ebenfalls meinen Freunden, die Verständnis für meine permanente Zeitnot hatten. Allen voran möchte ich meine Freundin Nicole erwähnen, die mich in schwierigen Situationen stets unterstützt hat. Sie sprach mir neuen Mut zu, trotz der Unmenge von Arbeit, die auf mich wartete. Ich bin allen Korrekturlesern (Kyle Barbour, Erin Leigh Flynn, Tasmina Tazeen und Olaf Thalmann) ebenfalls zu tiefstem Dank verpflichtet. Ich hoffe, dass ich mich irgendwann revanchieren kann. Ein letzter, aber besonderer Dank gilt meiner gesamten Familie. Sie haben mich selbstlos in all den Jahren unterstützt und mir stets Mut und Optimismus gegeben. Ohne sie wäre das alles nicht möglich gewesen.

Contents

Chapter 1

Introduction

"The important thing in science is not so much to obtain new facts as to discover new ways of thinking about them."

Sir William Bragg (1862 - 1942)

1.1 Motivation

Evolution, a fundamental and pervasive process, has always been an intensively debated topic, because the history of life on earth has left such an incomplete record. However, a wealth of research has provided new insights and methods to explore evolutionary questions across a variety of domains and species. Due to the relentless data explosion in biology and bioinformatics[1] and the increases in phylogenetic methodologies, we can now confidently generate phylogenetic trees for a diverse range of organisms.

These increases are one major reason for the surge of interest in comparative studies and the comparative approach in general. Nowadays, explicit phylogenetically based comparative methods (e.g., phylogenetic independent contrasts [18]) are the most general for asking questions about common or shared patterns of evolutionary change. They draw inferences about the rise of variation in character by analyzing interspecies variation. Usually, this involves testing evolutionary hypotheses on how these patterns may have evolved using statistical methods [1].

Ideally, but rarely, the full range of variation in a character is available to test the hypotheses. For example, a 'mouse to elephant curve' [3, p.215] is used in comparative studies which test the metabolic rate in different organisms [37]. Unfortunately often, only a limited range of variation is available and general implications of the study can be seriously affected by this. It is worthwhile to increase the variation of the character to have a broader spectrum of the entire variation, but for this, we must answer the question of which species warrant further study ('key species'). For example, we have to find species that extend the available spectrum of variation to strengthen the generality of the study finding.

[1] e.g., the GenBank release notes for release 162.0 (October, 2007) state that "from 1982 to the present, the number of bases in GenBank has doubled approximately every 18 months."

Comparative data can also reveal gaps in our understanding of biological features. These gaps, if systematically biased towards particular species or lineages, can generate noticeable statistical biases in comparative studies (e.g., see next section). It is common to read in write-ups of comparative studies that further sampling is needed, but no solution to identifying the most significant of these gaps has been given.

These different kinds of biases – *variation biases* and *gap biases* – can make a momentous difference to the conclusions one draws. Indeed, there have been cases where previous conclusions have been rejected or altered. One example is the work of Hamilton and Zuk [30], who studied the dependency of parasitism and mate choice in birds. Read and Harvey [69] reanalyzed this work by using a similar dataset and a more modern method. However, possibly the most important is that they included certain taxa to cover a broader spectrum of species. In summary, they did not find any support for the original hypothesis. There have also been cases where researchers resurrected hypotheses that were previously rejected (e.g., relationship between clutch size and egg size in birds [6], and clutch size and life span in Drosophila [61]).

Given the high costs of collecting data on organisms in the field and lab, the need to attenuate these biases becomes more and more paramount. At the present time, however, no systematic methodology exists for identifying these biases and key species, thus providing an impetus to develop such methods. Furthermore, by using phylogenetic comparative methods (like the method proposed in this thesis), one can quickly identify species that conflict with the general pattern of a feature or identify comparative patterns that previously lay hidden. Indeed, several exceptions have been found using this approach (e.g., Birkhead [5] revealed that some male birds suffered more from extra-pair copulations by male nonmates than others).

1.2 Introductory Example

One possible application could be envisioned by inferring the function of sleep. "Sleep is an evolutionary puzzle. Unlike ..., the functional benefits of sleep are not immediately apparent, and the costs of sleep appear to be substantial"[53]. In the last few years, bewildering gaps in our understanding of primate sleep have been uncovered. In a recent compilation of primate sleep data, for example, basic sleep patterns among the apes have been collected only in humans and chimpanzees [50][2]. Deciding data are thus missing for bonobos, gorillas, orangutans and gibbons. Only 20 species of primates have any data on total sleep times, and these data are heavily skewed towards terrestrial species: Only 18% of the primates are terrestrial, but 67% of the data come from terrestrial species [55]. Unfortunately, terrestriality has emerged only few times and this creates clusters of closely related species with sleep data. As visualized in Figure 1.1, this suggests that comparative studies of primate sleep may be systematically biased towards particular lineages. A reason for that could be the fact that the study of sleep is easier in terrestrial primates than in arboreal ones.

Data on sleep cycles (e.g., the phasing of REM-NREM sleep) are available for only eight primate

[2] http://www.bu.edu/phylogeny/about/index.html

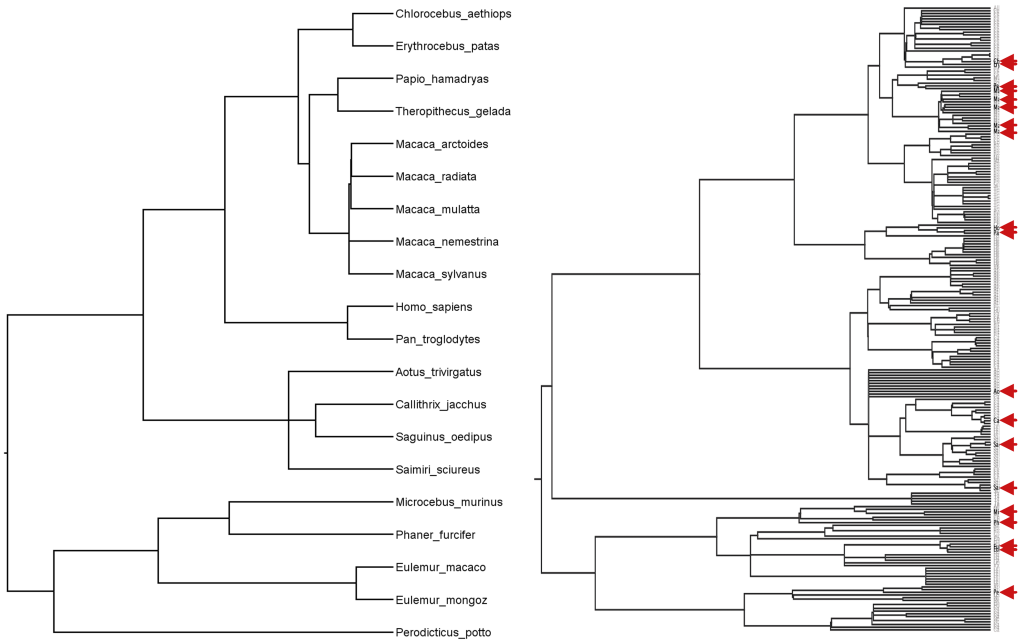


FIGURE 1.1: Example primate phylogenies. On the left side, a parsimony reconstructed phylogeny is shown with species where sleep data are available. On the right side, a more general primate phylogeny is shown [4]. All species from the left are highlighted with a red arrow, indicating that sleep data are available.

species [55], which is another illustration of a potentially biased data collection. These two examples highlight the importance of collecting sleep data on more primate species to test hypotheses for the evolution of this biological important trait. Given the costs of collecting such data, we need a way to systematically evaluate the species to study.

1.3 Subject of this Thesis

In summary, the idea for this thesis springs from the realization that for many of the most interesting questions in biology - such as the evolution and function of sleep - a variety of hypotheses (e.g., 'Predation risk influences total sleep duration.' or 'Sleep intensity increases with a decreased total sleep time.') have been proposed [15, 39, 40, 9], but exceptionally few species have been studied to test these hypotheses [50].

To test these hypotheses comparatively, we need to study more species - but which species should be studied? Indeed, this is the fundamental question of this thesis. By taking the hypotheses, comparative data relevant to the hypotheses, and a specific phylogeny, we can identify which species would provide the most compelling tests of hypotheses. We have developed a methodology that can directly address this decisive question. Using this method of *phylogenetic targeting*, gaps in our knowledge of particular lineages as well as in relation to particular biological traits of interest can be revealed. This is particularly important, because studying species is both expensive and

time-consuming. It is therefore extremely useful to select the species which should be studied given certain hypotheses, rather than collecting data by chance and hoping that it will be useful. More generally, *phylogenetic targeting* can help guide the increase in bioinformatics data into the most profitable directions.

Inspired by this *phylogenetic targeting* idea, the main goal for this project is to provide a new way to develop a systematic, quantitative and phylogenetic approach to identifying where future research effort should be placed. We also developed a computer program, *PhyloTargeting*, which implements this idea. It is web-based and freely available[3] and thus accessible for everyone with an internet connection. It provides a user-friendly interface, a variety of options to analyze the dataset, graphical visualizations of the results and many more powerful features that are helpful for probing a particular evolutionary question.

After collecting data for high priority species with the help of that new approach and computer program, the information can be made open to the public research community for further investigation. Thus, it is also a tool to get new insights into the process of evolution and into some of the most fundamental questions in organismal biology.

1.4 Organization of this Thesis

This thesis systematically describes the new methodology and the developed computer program in detail. The remainder of this document is organized as follows: After the introduction, Chapter 2 explains important background knowledge and principles that are crucial for understanding the idea of the approach. Chapter 3 discusses related work. Then, in Chapter 4, a detailed description of the *PhyloTargeting* approach is presented. The developed algorithms are shown, as well as the methodology itself. Chapter 5 discusses efficiency issues and analyzes the program and its data structures. Chapter 6 applies a real-world dataset to the *PhyloTargeting* framework, to highlight the practical usefulness of the approach. For this, a sleep dataset on primates will be used to test hypotheses about the evolution of sleep. Chapter 7 discusses general issues and summarizes the ideas and the implementation. The Appendices provide supplementary material, which is helpful for the interested reader. Finally, references of this thesis are listed.

[3] http://www.bioinf.uni-leipzig.de/~achristian/

Chapter 2

Background

"The comparative approach is not new. Indeed, it was Darwin's favoured technique. ... In short, comparative studies have taught us most of what we know about adaptation."

Harvey and Pagel (1991), page v

2.1 Comparative Biology

2.1.1 Testing Hypotheses

One of the most frequent questions in comparative biology is whether different features are correlated, thus suggesting that there exists an evolutionary process linking these features [33]. This can be due to other aspects of the organisms or their phylogeny. Numerous examples are possible using the example of sleep: Is sleep duration correlated with body mass? Does sleep intensity decrease in species that experience greater risk of predation? Does REM sleep duration covary with memory needs? Such questions of trait correlations usually address an underlying adaptive hypothesis. Numerous methods have been proposed to address such correlation issues in a comparative context, and the assumptions they require differ greatly. For example, some of the methods assume a specific model of evolution (e.g., Brownian motion in the phylogenetically independent contrast method [18]), some assume that the phylogenetic topology as well as the branch lengths are known and error-free (e.g., also phylogenetically independent contrasts), some assume that within-species variation is negligible (e.g., Grafen's regression [26]) and other methods require that ancestral states have been accurately reconstructed [43]. These assumptions usually increase the power of the method, but they have a significant drawback: If the assumptions are unrealistic or too strong, they may limit the applicability of the method. If the assumptions are not met and the method is nevertheless applied, this may lead to erroneous results [44].

We want to highlight and describe two particular methods in the following sections, because they are decisive for the *PhyloTargeting* program. Although they are, in their original sense, not suited to determine key species, we will use a combination of these methods in this thesis.

2.1.2 The Pairwise Comparison Method

The idea of pairwise comparisons is not a new one- its roots can be traced back at least as far as Salisbury 1942 [73], and even Darwin used species pairs for analysis. Although newer methods are available that make better use of the variation, the method of pairwise comparisons is actively used in organismal biology ([18, p.13]; [52]; [64]; see also [70] for a closely related approach). Frequently, the method is used to test correlated trait evolution.

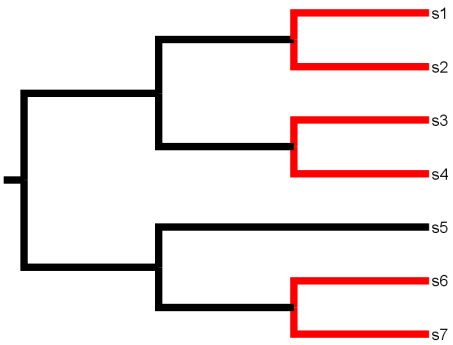


FIGURE 2.1: In this hypothetical example, a sample phylogeny is shown, consisting of seven species. By using sister species pairs, three pairwise comparisons (highlighted in red) can be selected.

This method uses species pairs and their contrasts (which represent differences between the trait values) on a phylogeny to test hypotheses based on differences in traits within each pair. Different species pairs must be phylogenetically separate, and thus they can be seen as independent data points that can then be applied to statistical analysis. Many researchers recognized that comparisons of closely related species are particularly valuable, but what is the reason for that? In most cases, other factors may influence the trait of interest (confounding variables). If these factors are shared among relatives, which is assumed to be true, then the use of these congeneric pairs provides a way of holding these factors constant within each comparison. That enables to test the role of another factor that exhibits less signal, and has diverged between close relatives. Thus, fewer confounding variables can be expected to influence the result.

In practice, one identifies congeneric species pairs on the phylogeny, and calculates their difference. Each such contrast then represents an independent data point. As Felsenstein [19] noted, on a dichotomous tree, the maximum number of possible contrasts is equal to the number of species in the tree, divided by two, rounded down to the nearest whole number (the rounding is due to the fact that for phylogenies with an odd number of species, one species is left and cannot be paired).

The method is attractive for its reliance on relatively few assumptions, and the advantages compared to other comparative methods such as phylogenetically independent contrasts (see next section) are as follows:

- There is no requirement to reconstruct ancestral states, only the data from the tips of the tree are used.

- It works well with any kind of data: continuous variables, and especially discrete ones.
- It compares closely related species instead of distantly related ones and thus helps to control for confounding variables.
- It is more robust to phylogenetic uncertainty: Errors arising from phylogenetic uncertainty are minimized because sister species are favored over more distant relatives [52]. These pairs share only a small number of branches, and thus, uncertainty is decreasing, because potentially erroneous reconstructed nodes near the root of the tree are not used.
- It does not rely on explicit evolutionary models.
- Pairs of species can be targeted that offer the most power to test hypotheses (see also Chapter 4).
- It is also very useful when equivalent data cannot be collected across the tree (see also Chapter 4)

However, there are some important drawbacks that are crucial to be mentioned. They will be intensively discussed in Chapter 7.

2.1.3 Phylogenetically Independent Contrasts

In 1985, Felsenstein published the first phylogenetic statistical method for the analysis of comparative data [18] that has no assumptions on either the topology or branch lengths. It can be seen as a breakthrough contribution, because between 1985 and 2002, it has been cited 1462 times [10]. His methodology, often referred to as phylogenetically independent contrasts, can be used to test hypotheses in a comparative context. By taking the species phylogeny into account, it guarantees statistical independence among the data points. It can be seen as a generalization of the presented pairwise comparisons method, because it considers the divergences that have occurred at each bifurcation (that is, also ancestors can be compared) in the phylogenetic tree, instead of using only species divergences. Thus, every branch of the tree is used and more phylogenetic information can be incorporated in the analysis (see Figure 2.2). In practice, if we are given a tree with n species, congeneric pairs of species are contrasted. The direction of subtraction is arbitrary. Usually, however, the main hypothesis is commonly forced to be positive [67, 51]. After pruning these pairs from the tree, the ancestral nodes are estimated using a weighted mean of the values from the descendants. The process starts again, and thus further contrasts are computed involving the values estimated for internal nodes. All contrasts are divided by the square root of the sum of their branch lengths (standard deviation) to give them a common variance as required by most statistical tests. Finally, $n-1$ contrasts are computed that can then be used for regression or correlation analysis. As Garland [23] noted, statistical tests must be computed through the origin.

The method has some important assumptions:

- The phylogenetic topology is known and assumed to be correct.

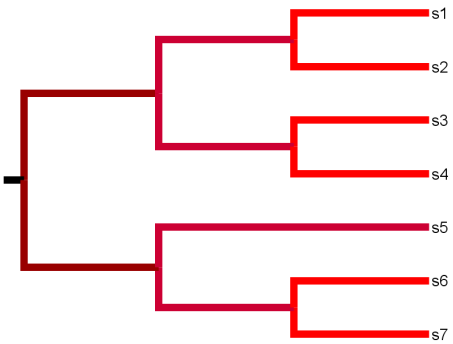


FIGURE 2.2: The same phylogeny as shown in Figure 2.1, but with the phylogenetically independent contrasts method. Instead of using only sister species, all branches of phylogeny are used (six in total, indicated by different colors). See text for details.

- Branch lengths are assumed to be in unit of expected character change.
- Within-species variation does not exist or is negligible.
- The process of evolution is assumed to be a Brownian motion process.

Despite these drawbacks, it has proven robust over a number of studies and simulations, and it is the most commonly used method for testing adaptive hypotheses in organismal biology (see [10] for a detailed list). Until today, there have been numerous adaptations of this method (e.g., see [25] for an overview). Some ideas of these methods will also be used in our approach as described in later chapters.

2.2 Mesquite and the NEXUS File Format

"Mesquite is software for evolutionary biology, designed to help biologists analyze comparative data about organisms. Its emphasis is on phylogenetic analysis, but some of its modules concern population genetics, while others do non-phylogenetic multivariate analysis"[1]. Mesquite [45] includes a variety of analyses, including tests for correlation and character evolution, ancestral states reconstruction using parsimony and maximum likelihood, and tests of speciation and extinction rates.

[1] http://mesquiteproject.org/mesquite/mesquite.html

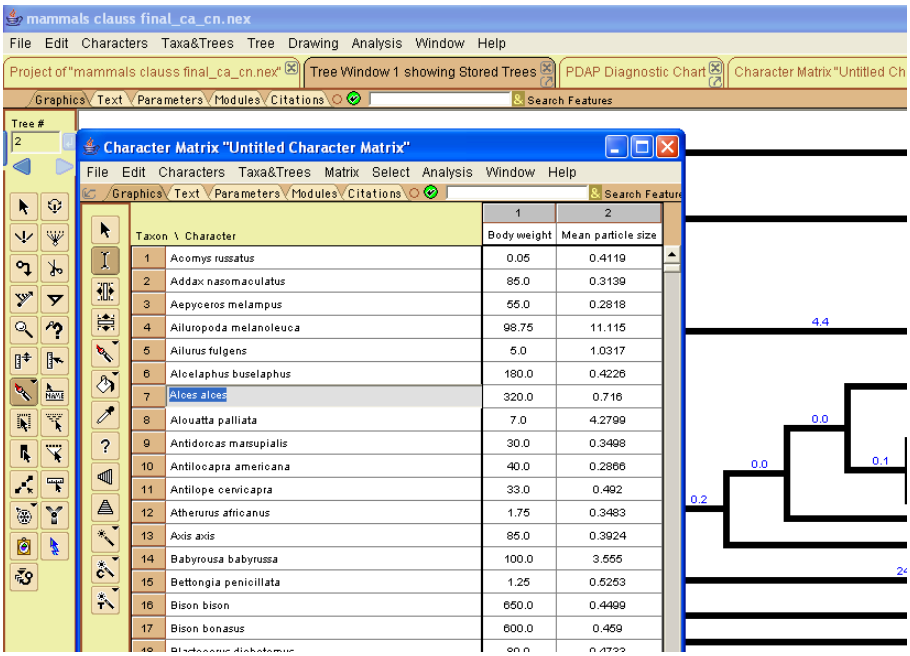


FIGURE 2.3: A sample screenshot from Mesquite. In the front, a character matrix window is shown; in the back, a part of the phylogeny is illustrated.

It is appropriate for comparative analysis, and supports different comparative methods such as Felsenstein's phylogenetically independent contrasts or the pairwise comparisons method as described by [44] (see Chapter 3).

Mesquite is widely used, and it works with the NEXUS format, which is a "file format designed to house systematic data" [42]. The primary design feature is modularity. It is composed of a number of different blocks (e.g. TAXA or TREES), and the four main principles are as follows[2]:

1. **Expandability**: New blocks or statements within existing blocks can be added with minimum disruption. Because of the structured design, programs are able to recognize relevant elements and disregard irrelevant ones. New elements are thus just ignored if they are not supported.

2. **Inclusivity**: The file can contain all the information a researcher is interested in, including character data, morphological data, assumptions, trees, and so forth.

3. **Portability**: The file format can be used on every operating system, since only simple text files are used. Moreover, the insensitivity to specific newline characters allow the use of different operating systems.

4. **Processibility**: Programs which are able to read NEXUS files can pick up the information they desire, and skip commands or blocks they do not need or support.

[2] taken from the NEXUS paper

The NEXUS format has many functions. Currently, no program implements all of the features NEXUS provides.

2.3 Applied Informatics

PHP

In this section we give a brief introduction of the programming language PHP (a recursive initialism for PHP: Hypertext Preprocessor). The following citation clearly describes the nature of PHP: "PHP is a widely-used general-purpose scripting language that is especially suited for Web development and can be embedded into HTML"[3]. Such a language should have criteria as follows to be usable:

1. Fast prototyping and implementation
2. Support for modern programming paradigms
3. Scalability and performance
4. Interoperability
5. Extensibility

Especially the first criterion has always been a strength of PHP, and with version 5 at latest, "PHP has fully embraced the rest of these ideas as well"[74, p. 2]. It provides an immense number of extensions (such as PDF creation, database access, and remote services), and some of these extensions and external libraries are used in the *PhyloTargeting* program. Additionally, powerful features for rapid web development are available, and it offers enhanced security, although security is more a matter of the programmer than the programming language itself. Moreover, it provides a session management that allows for storing of data between pages. Together with possibilities for fast and easy serialization, it also offers a way to handle and save the state of an application.

Phylogenetic trees and their computer representation

Phylogenetic trees can be represented in a computer as a special rooted, acyclic, directed graph. We now introduce these terms to define the concept of a tree. Moreover, these terms are important for other chapters.

Definition (directed graph): *A directed graph G is a pair (V, E), where V is a finite set and E is a relation on V. The elements of V are called nodes, and the elements of E are called edges.*

Definition (predecessor, successor): *u is a predecessor of v, and v is successor of u in G if (u, v) is an edge of G.*

[3] http://www.php.net/

Definition (path): *A path in G is a sequence $v_1, \ldots, v_n$ of nodes such that v_i is a successor of v_{i-1} for $i = 2, \ldots, n$. The path is a cycle if $n > 1$ and $v_n = v_1$.*

Definition (acyclic): *A directed graph is said to be acyclic if it contains no cycles.*

Definition (rooted graph): *An acyclic graph is said to be rooted if exactly one of its nodes, called the root, has no predecessors.*

Definition (tree): *A rooted, acyclic, directed graph is called a tree if each of its nodes, excluding the root, has exactly one predecessor and none or at least two successors.*

Thus, phylogenetic trees can be represented in the computer as a set of nodes and a set of edges. In the basic implementation, each node maintains a list of references to all incident edges, and each edge maintains a reference to its source node and to its target node as well as its length. Of course, additional attributes can be specified.

Definition (last common ancestor): *The last common ancestor is defined between two nodes v and w as the lowest node in a tree T that has both v and w as descendants (where we allow a node to be a descendant of itself).*

In other words, the last common ancestor is the shared ancestor of v and w that is located farthest from the root.

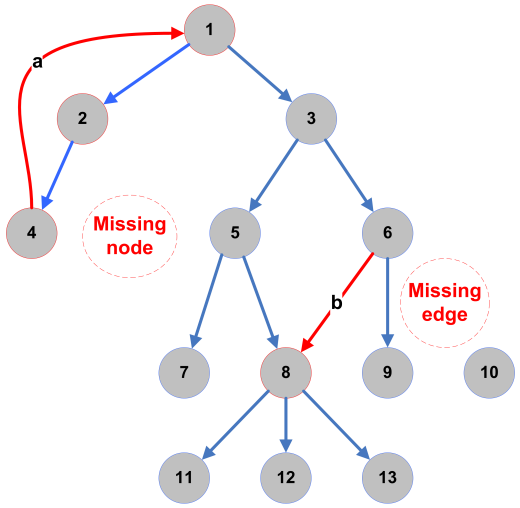


FIGURE 2.4: A sample graph that is no tree. It summarizes the basic terms and every red element presents a violation of the above stated definitions: The cycle $1 - 2 - 4 - 1$ can be removed by deleting edge a (marked in red at the left). One node (2) has only one successor, which can be changed by introducing a new node as successor or creating an edge from node 2 to node 10. Furthermore, node 8 and node 10 do not have exactly one predecessor: Node 8 has two, node 10 zero. This can be solved by deleting edge b (marked in red) and adding an edge from node 6 to node 10.

Tree properties

Trees are ideally dichotomous. In reality, however, phylogenetic trees often contain some uncertainty, leading to polytomous trees. As we will later see, these trees have to be treated special in

some of the developed algorithms. The two kinds of polytomies are as follows:

- **Hard polytomy**: The hypothesis that a common ancestral population split through cladogenesis into multiple lineages, e.g. in the Cambrian explosion. Thus, they may indicate true simultaneous speciation events.

- **Soft polytomy**: This reflects uncertainty in which resolved pattern is the best hypothesis and the more common intended meaning of a polytomy. Thus, one is not really expecting that the same ancestor gave rise to all daughter taxa, but lack of knowledge prevents representation of a more detailed speciation process.

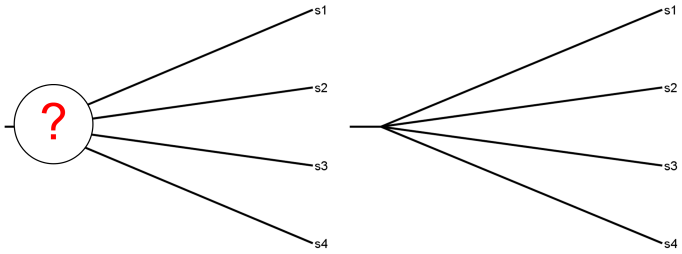


FIGURE 2.5: Hard and soft polytomies. On the left side, a soft polytomy is shown, on the right, a hard polytomy. See also text for details.

Chapter 3

Related Work

"An approximate answer to the right problem is worth a good deal more than an exact answer to an approximate problem."

John Tukey (1915 - 2000)

3.1 Overview

To the best of our knowledge, the issue of systematically targeting key species has not yet been systematically addressed in the literature. Several individuals did recognize the necessity of identifying target species [79, p. 6] [78, p. 15], but only guidelines for this selection process have been given. Moreover, these guidelines are specific to the question of interest.

In this thesis, a variant of the presented method of pairwise comparisons is used. It has been well studied and is frequently used in the scientific community; however, almost always, the method has been chosen to address evolutionary hypotheses, instead of the issue we addressed above. Therefore, these cases cannot be considered as similar approaches, although they also use this methodology.

3.2 The Approach of Maddison

Surprisingly, one approach is worth to be mentioned. Maddison presented a methodology for choosing pairs using the pairwise comparisons method on a phylogeny [44] (we will neglect further citation of this paper throughout this chapter). Even though he did not directly address the question of identifying key species, he formulated a similar question: Each pair should satisfy a criterion of relevance in a way that "it should represent a comparison relevant for the question of interest". Although his idea is more theoretically based, it can be seen as a preliminary version of identifying key species, because he noticed that not each pair can contribute to or against hypotheses. The necessity to find compelling pairs in the pairwise comparisons approach has been noticed even before: For example, Read and Nee [70] noticed that pairs contribute only for or

against a hypothesis of association if they differ in at least one character. However, to the best of our knowledge, no systematic and quantitative approach that addresses this particular question exists.

But what criteria should be used to choose a particular pairing (see also Figure 3.1)? In the case of continuous characters, every pairing could be useful, because it is likely that the difference between two species differs from zero. For two discrete characters, however, Maddison considers three different cases, which can be discriminated in the number of characters that differ between the species of a pair (see the Maddison paper for more details):

- No regard to the states of the characters: All possible pairings[1] are considered, and the pairing that maximizes the number of pairs is chosen.
- Pairs that contrast in one binary character: Only pairs are chosen that contrast in one binary character.
- Pairs that contrast in two binary characters: Only pairs are chosen that contrast in all two binary characters.

After generating all possible pairings that satisfy the condition stated above (using dynamic programming algorithms), statistical tests are applied for each such pairing and a table presents the range of significance values. Thus, dependence on arbitrary choices of the pairings is eliminated through generation of all acceptable pairings and their significance. This methodology has been implemented in Mesquite, and an example screenshot is given in Figure 3.1.

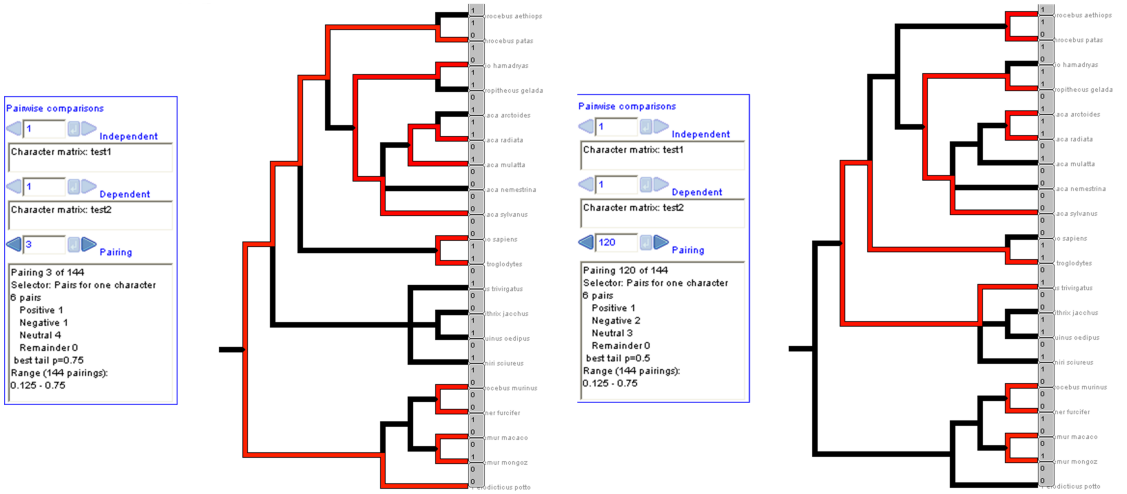


FIGURE 3.1: Sample output in Mesquite: An analysis of pairwise comparisons automatically chosen by the algorithm of Maddison (2000).Two different pairings are shown, and only pairs that differ in one trait are considered. The left one is described in more detail. However, the same principle applies for the right one. There are 144 pairings of terminal taxa with 6 pairs contrasting the independent (categorical) variable. The first pairing (#1) is shown: 1 pair is positive, 1 is negative and 4 are neutral (not significant with p=0.75). Among all 144 pairings, none are significant (ranging from 0.125 to 0.75).

[1] see the next chapter for a definition of a pairing

However, this approach has some significant drawbacks compared to the goal we want to address:

- It allows only discrete traits.
- Only one main hypothesis (called independent) and one alternative hypothesis (called dependent) are supported.
- Neither additional information about the calculated trait differences for each pair is provided, nor is a score calculated that represents how informative this particular pair is.
- Only three different methods for calculating these pairings are available, and they are too specific.
- It is not clearly arranged, especially with large phylogenies.
- Only automatically calculated pairings are shown, the investigator cannot manually select particular pairs.

Thus, we are limited in what we can investigate in the pairings and this approach is not capable of identifying where future research effort should be placed. However, the basic idea is comparable, and we will extend it in the next chapter.

Chapter 4

Phylogenetic Targeting

"The mere formulation of a problem is far more essential than its solution, which may be merely a matter of mathematical or experimental skills. To raise new questions, new possibilities, to regard old problems from a new angle requires creative imagination and marks real advances in science."

Albert Einstein (1879 - 1955)

The purpose of this chapter is twofold: First, to show the biological methodology to accomplish the task of identifying key species, and second to present the informatics approach in more detail.

4.1 Modeling

4.1.1 Programming Language

The first crucial decision was to choose among numerous programming languages. Since a web application should be developed and accessible from the internet, a number of programming languages can be excluded from the outset. Another requirement is that it can be downloaded and used by as many users as possible. This constraint implies that a widely used, cross-platform, modern, and easy to install programming language should be used. Only *PHP*[1] and *Perl*[2] remain as plausible options after considering these requirements. Perl, the dominant programming language in bioinformatics, and PHP, one of the most widely used web programming languages, are both appropriate for the developed application. PHP is extremely easy to install; in general, only a local web server, e.g. Apache, is required. Precompiled packages, such as *XAMPP / LAMPP / MAMP* (depending on the operating system), exist which help to simplify the installation process (see Appendix A). Additionally, PHP is better suited for web applications. Therefore, we chose PHP.

[1] http://www.php.net/
[2] http://www.perl.com/

4.1.2 Data Structures

It is apparent what data structures a phylogenetic related program such as *PhyloTargeting* should use. Because most of the calculation is based on an underlying phylogenetic tree (e.g., traversing, looking for nodes, selecting subtrees, allocating branches), it is appropriate and efficient to use a specialized tree data structure. The way it is implemented is as described in Chapter 2, however, a new type of object must be embedded to represent the complex interaction which is needed in the application: pairwise comparisons, which are similar to edges (more details can be found in later sections). Indeed, they can be seen as connections between leaves, representing a comparison between these two species. They do not have an explicit length or weight like regular edges, but they do have a score, which is likewise in our cases. Thus, two kinds of edges are used: one classic type that represents the connection between the nodes and the other kind representing connections between leaves of the tree, the pairwise comparisons. This is comparable to a visual representation of RNA secondary structures, the *secondary structure graphs*, because they also incorporate two different kinds of edges.

4.1.3 Class Overview

PhyloTargeting uses the object-oriented paradigm, which provides a clear modular structure for programs. Therefore, it is appropriate for defining abstract data types. Implementation details are hidden, and each unit has an acutely defined interface.

To summarize the discussed aspects, a tree object consists of node, edge, and pairwise comparison objects. The following UML diagram shows in more detail how these objects are represented internally and how they interact.

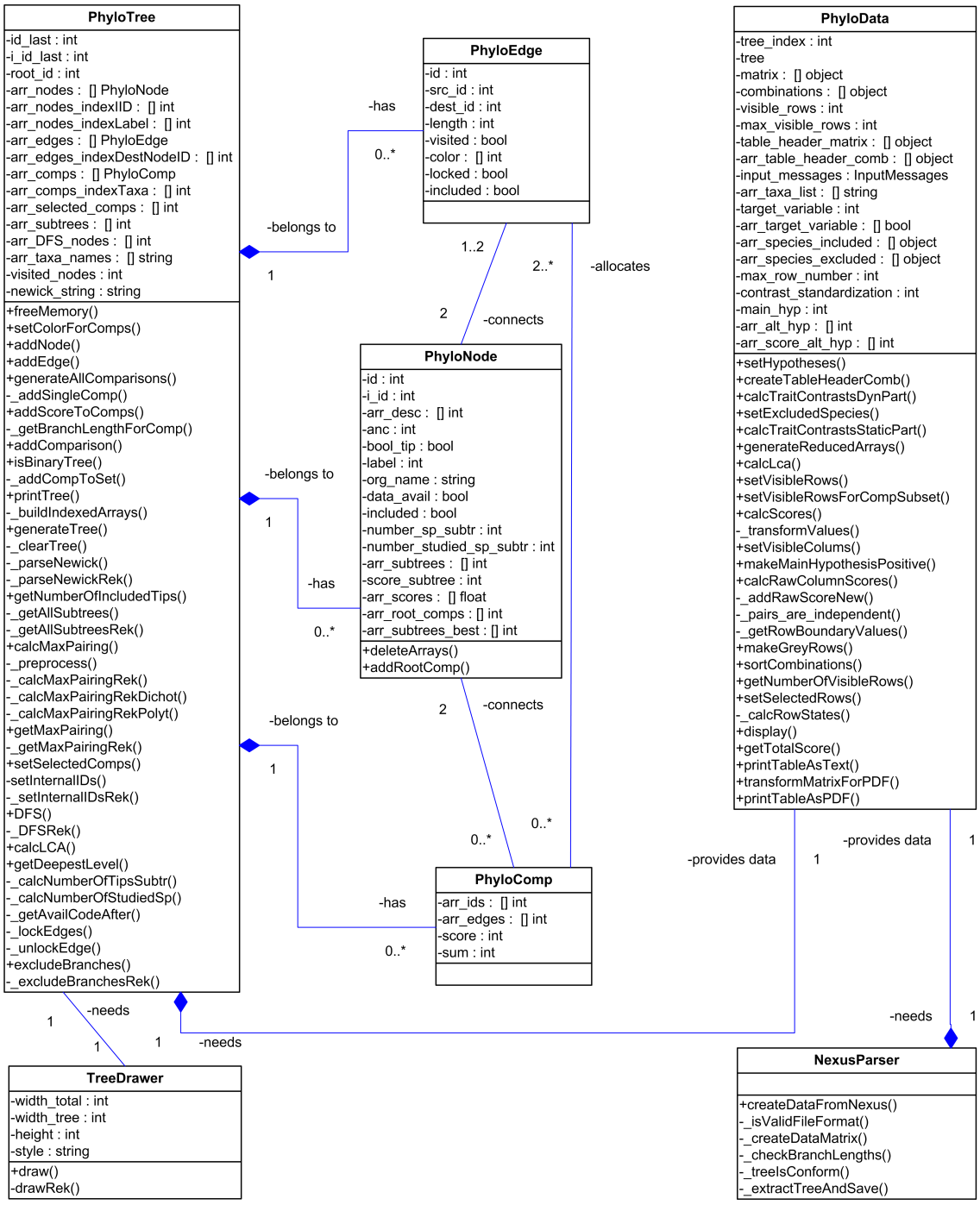


FIGURE 4.1: UML Diagram of the main classes that are used in the application. The diagram is simplified, because only important and meaningful functions are listed (that is, especially all *set* and *get* methods are excluded), and no details for functions (e.g., return type, parameters) are provided for simplicity.

4.1.4 Workflow Overview

Although the terminology and the interaction among different processes have not been introduced yet, a typical workflow is presented that helps to develop a basic understanding. Such a workflow would be as follows, and the most important of these features are also shown in Figure 4.2 and described throughout this chapter:

- The investigator uploads a NEXUS data file containing multiple species, a corresponding tree in the NEWICK format and at least two traits representing the hypotheses of interest.
- Then, a setting must be specified on which the calculation is based, which consists of the following elements:
    - A selection of the species that should be included in the analysis
    - A phylogenetic tree on which the calculations are based
    - A main hypothesis that reflects the question of interest
    - Optionally, one or more alternative hypotheses that reflect potentially confounding variables or alternative explanations that the user might wish to investigate
    - For each alternative hypothesis (if specified), a scoring mechanism must be declared that specifies how the maximum score is assigned, specifically with regard to whether change in this variable is minimized or maximized (and the direction of maximization).
    - An optional target variable that screens all pairwise comparisons for the most informative ones; typically this will include details on the species that have been studied with respect to some dependent data (e.g., EEG data on sleep).
- With this information, the calculation of all pairwise comparisons and their information content relative to the specified hypotheses is possible. This information content represents how compelling these species pairs are for further data collection.
- After calculating all of the mandatory elements, the following options are possible:
    - A summary table can be examined in step 3. This table lists the most important features, and is a good starting point for further investigation.
    - A pairing can be specified either manually using the *contrast selection* feature, or automatically using the maximal pairing algorithm.
    - Particular species or the pairing itself can be examined separately using the options from the analysis step.
- Different graphical visualizations and export options are provided for further investigation (e.g., the distribution of all pairs in the tree, or export to common file formats, such as PDF or comma-separated text files)

- The application state can be saved to continuing the analysis at a later date if desired.

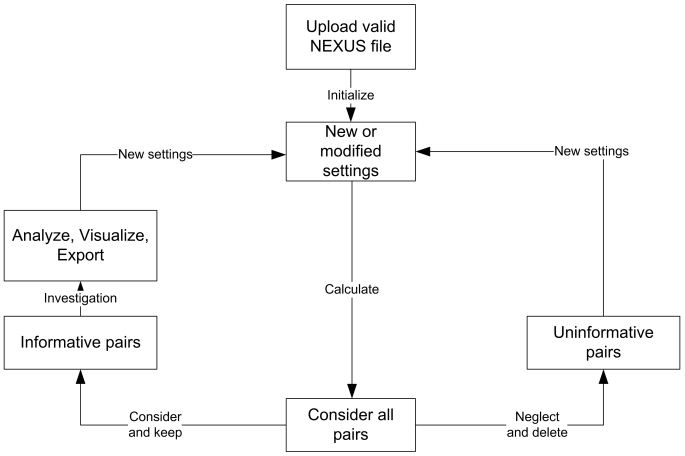


FIGURE 4.2: Typical workflow in the *PhyloTargeting* application.

4.2 Initialization

The first step is a proper initialization, which includes loading an appropriate data file and specifying a set of settings that contains all mandatory information needed to do the calculations. These initial steps are described briefly in this section.

4.2.1 Loading Data Files

The investigator must either upload a valid data file to the program, or he may use the provided example file. Here, we briefly describe which data files can be used; further details, including a screenshot of an example file, are provided in Appendix B. The data file must be in the NEXUS format, and NEXUS files from other programs may have to be modified. For example, this can be achieved by importing and saving them in the program Mesquite, because Mesquite data files are known to have full compatibility. If continuous data are involved in the analysis, Mesquite will also be a helpful tool to enter the data into the appropriate format.

If the loading was not successful due to errors in the data file, a detailed description of the nature of the error is provided, and moving on to the other steps is prohibited.

4.2.2 Settings

After successfully loading the data file, the investigator is able to move on to step 2. Here, some settings have to be made which are needed for the calculation. These settings are as follows, and detailed explanations are provided in the next sections:

- The investigator chooses either all or an arbitrary subset of species that should be considered in the calculation. All excluded species will also be excluded completely in the analysis. More details are provided in section 4.3.3.

- A tree must be selected on which all information are based. *PhyloTargeting* automatically extracts all tree definitions from the data file, and thus more than one tree can be selected. Indeed, different trees can yield to different results, and for this reason, the underlying phylogenetic tree is a very important component of the whole analysis.

- Hypotheses and scoring mechanisms must be chosen. This reflects the underlying question that the investigator wants to address and is explained in full detail in section 4.3.5.

- An optional target variable and a constraint can be specified. This feature is explained in section 4.3.6.

- The investigator can choose if pairs should be standardized or not (see also section 4.3.5).

After the user specifies these settings, the program does some basic calculations that are called whenever new or modified settings are applied to the program. These calculations include the following:

- Species that had to be excluded due to missing information

- The number of pairwise comparisons that are informative

After confirmation by the investigator, the calculation starts. A progress bar [8] indicates the amount of progress.

4.3 Basic Methodology and Algorithms

4.3.1 Overview and Hypotheses

Comparative tests can be roughly divided into two groups: First, broad-scale approaches (e.g., [71]), which test hypotheses across many species, and second, focused comparisons consisting of only few species [77], which allow the user to tailor the test to the species of interest. Both approaches have advantages; for example, broad-scale comparisons provide a means to assess the generality of a pattern, while focused comparisons provide a means to collect more detailed data on several species. However, both methods also have disadvantages. In the broad-scale variant, a major constraint is that data must be collected for as many species as possible, and the data must be comparable across species. Unfortunately often, this is a rather unrealistic assumption, because data collection is an expensive and time-consuming task and methods must be tailored to the species being studied. Usually, data are available for only a fraction of the species in a clade, and one cannot get the desired data for all species that have not been studied. The main drawback of the pairwise comparison approach is that it is difficult to test alternative hypotheses due to an insufficient number of contrasts.

In summary, the broad-scale and pairwise approaches can be seen as two extremes, and an approach that combines the advantages of both is needed. That is, we need to select a set of species that offer the most power to test hypotheses, and all of them should be phylogenetically separate.

Usually, two hypotheses are tested against one another: One main hypothesis and one alternative hypothesis. Hypotheses can be mutually exclusive, which means that the occurrence of any one of them automatically implies the non-occurrence of the remaining one, so that only one hypothesis can be true. However, hypotheses can also be non-mutually exclusive (sometimes also called compatible), which means that two or more of explanations for some pattern are possible. The occurrence of one does not prevent the occurrence of the others in all cases. Sometimes also more than two hypotheses have to be tested, either to control for potentially confounding variables, or because there are truly multiple hypotheses that need to be considered. All discussed scenarios can be applied to the *PhyloTargeting* framework, and thus the program offers great flexibility, depending on the questions at hand.

A main aim for this thesis is to develop a hypotheses-driven approach that is able to identify comparisons that offer the strongest tests of one or more hypotheses. But how can they be identified? We will present some main ideas using a sleep dataset to test hypotheses related to the function and evolution of sleep as an introductory example[3]. The goal of this example is to develop a basic understanding how one can address this question.

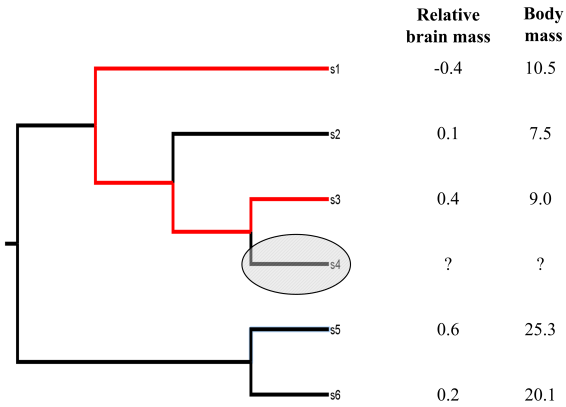


FIGURE 4.3: Example phylogeny consisting of six species and two traits that can be screened using the *PhyloTargeting* approach. Species for which no data are available are indicated with an ellipse. Among all possible pairwise comparisons, the one between species *s1* and *s3* has the most power, because albeit both have a similar body mass, the difference in the relative brain mass is big. See also text for details.

After collecting data for the species of interest and indication of species where data are not available, one can construct variables that reflect the hypotheses on which the calculations are based. In this example, we test the hypothesis that the major function of sleep is related to the brain. Specifically, we constructed a variable that reflects the residuals from the phylogenetically-adjusted regression line between brain and body mass, which we call *relative brain mass* (see also Appendix C). Relative brain mass reflects brain size after controlling for body mass by using measurements from the captive individual whose brain was measured. Yet we might also be

[3]These hypotheses are also used in Chapter 6.

interested in body mass as an ecological variable, as this can influence predation risk, diet and life history (all of which have been proposed to affect sleep times). We thus control for body mass by including data on body mass in wild animals as one of the alternatives. Our goal is to identify species have similar body mass, but big differences in relative brain size. To identify these key species, we generate pairwise comparisons that enable us to test these hypotheses by initially generating all possible pairwise combinations. We can now find species pairs that enable us to pit the hypotheses against one another by scoring them in relation to the settings.

By applying this procedure, we can identify the species that can be compared to test predictions (which usually are derived from a theory). Moreover, we can put them in competition against one another to identify 'high priority' species for future data collection. In the example above, species from the highlighted pairwise comparison (*s1* and *s3*) will be identified as key species, because they have the most power to test the hypotheses.

4.3.2 Pairwise Comparisons

Different methods are imaginable for identifying these key species. As presented, the method of pairwise comparisons is well-suited to apply to the question of the thesis. A major reason is that some species are not directly comparable: For example, some cetaceans sleep with only $\frac{1}{2}$ of their brains, thus making it difficult to compare the measurements of sleep to other mammals - one would only want to compare cetaceans with other cetaceans. Or, an experiment based on cognitive skills might be appropriate for only some of the species in the lineage; one would have to tailor the experiment to different species, and to only compare those species given the same experiment. In the original method (assuming that the phylogeny has n distinct species), only sister species form a pair, and thus a maximum of $\frac{n}{2}$ pairs, rounded down to the nearest whole integer, can be formed. However, in this approach, we theoretically allow any pair of species, independent from their phylogenetic position; in other words, we do not have to rely on sister species only. This is indeed useful for our purposes, because pairs of non-sister species could be compelling enough to test hypotheses (see also Figure 4.3). This cannot be ruled out and therefore, all possible combinations have to be systematically generated initially. This methodology implements some ideas of the Maddison [44] approach (see Chapter 3), however, it is far more general and eliminates the main drawbacks of his implementation. Maddison proposed that "the algorithms presented here can be considered only the beginning of a suite of alternative algorithms that could eventually be derived". Indeed, the presented methodology is actually both a derivation and also a generalization that addresses a different question by using similar methodologies. As we will see later, it contains elements of the phylogenetically independent contrasts method as well, and can thus be seen as a combination of both.

Generation of all pairwise comparisons The procedure that generates all possible pairwise comparisons is trivial, and thus only a brief description is given. We parse the input file, generate a table consisting of all species including their trait values, and then loop over this table to engender all possible pairwise comparisons. Each species can pair with each species, except with itself. This yields the following equation for the maximal number of possible pairs:

$$N_{pairs} = \frac{n(n-1)}{2} \tag{4.1}$$

Each of them has to store some additional information for a variety of reasons, and the most important ones are listed as follows:

- The two species that form the pair
- Information about its state
- Trait differences for all specified traits
- Scores
- Phylogenetic information

The first bit of information is clear, and the third, fourth and fifth are discussed separately later. However, we give some details about the second bit of information.

States of a pairwise comparison It is not sufficient to simply generate all pairwise comparisons, for the purpose of our approach, a way to declare different states to a particular pairwise comparison is also a necessity (e.g., this is mandatory for the visualization, see section 4.3.7 and section 4.4.1). The following list explains and describes these states, and how they interact.

1. Active and passive pairwise comparisons

   Active comparisons are simply pairwise comparisons that are included in the calculation, and that fulfill the following three conditions:

   - Both species are in the list of the species that should be included.
   - No missing values in either the main hypothesis or one of the specified alternative hypotheses are detected.
   - If a target variable constraint is specified, then this value must be valid according to the specified settings in step 2.

   Therefore, passive ones are pairwise comparisons that violate at least one of these conditions, and they are not included in any calculation unless new settings are submitted to the program.

2. Different states of an active comparison

   The following states can be assigned to an active comparison:

   - *regular* (this is the default state; a priori, all pairwise comparisons are in this state)
   - *selected* (pairwise comparisons that are determined automatically by the maximal pairing algorithm or pairwise comparisons that are selected manually using the contrast

selection feature in step 3)

- *phylogenetically non-separate* (all pairwise comparisons that share at least one branch with at least one selected pairwise comparison are automatically assigned to this state)

Whenever a pairwise comparison changes its state, this may affect the state of other pairwise comparisons.

However, it has to be mentioned explicitly that these states are bounded to a specific settings as specified in step 2.

4.3.3 Missing Information and Screening Measures

In comparative biology, researchers often face the problem of incomplete data sets. Almost always, data for a particular trait are available for only some species, and even in those information about other traits can be partially missing. A fundamental question is how to handle pairwise comparisons that contain missing information, an issue that will be discussed now. Furthermore, screening measures are summarized.

Missing information Whenever an investigator specifies a setting (hypotheses, a set of species, scoring options, an optional target variable), the following procedure is applied: All possible pairwise comparisons are generated initially and a function is called that determines which pairs can be neglected. That is, all species that have missing information of any kind in one of the traits reflecting the hypotheses (either the main hypothesis or one of the alternative hypotheses) are ignored, because unless the investigator changes the settings, those species or pairs of species are uninformative. This is due to the fact they the power of such a pair cannot be determined if trait values are missing. Other constraints will be discussed separately. However, the investigator is informed if the application determines any kind of missing information.

If the investigator changes the settings, the procedure starts again. Thus, it is guaranteed that every useful pair is considered whenever new settings are provided to the program. This measure is done automatically and independently of the settings.

Screening Measures Mechanisms are needed that screen the pairwise comparison space for informative and compelling pairings, because the number of possible pairwise comparisons grows quadratically with the number of species and thus becomes large rapidly. The following selection criteria summarize which species pairs are excluded, and this is also visualized in Figure 4.4:

- Pairs that have at least one species that is excluded by the user.
- Pairs that do not fulfill the target variable constraint (see section 4.3.6).
- Pairs that contain any kind of missing information in relation to the hypotheses.

With these measures, the space of all pairwise comparisons can be greatly reduced, which is mandatory and useful if large data sets are chosen. Hence, only informative pairs are considered and displayed and non-informative pairs do not bias the analysis.

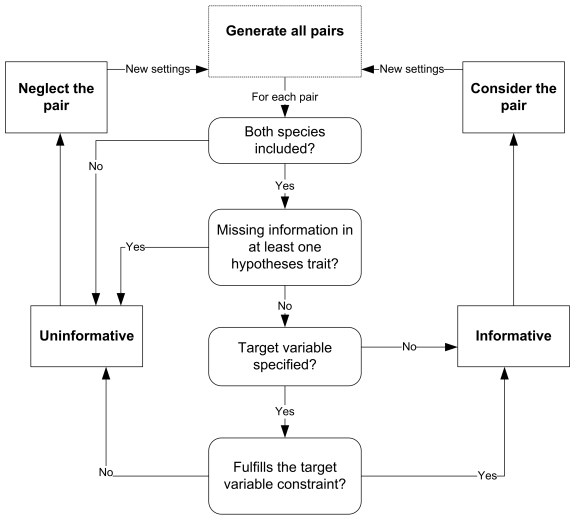


FIGURE 4.4: Screening measures after generating all pairs. See text for details.

4.3.4 Phylogenetic Information

In order to calculate a score for each pair, we need to incorporate phylogenetic information. Phylogenetic relatedness must be considered, since more closely related species will have genes or traits in common through descent from common ancestors and are thus "not solely a product of their current environment" [10, p. 54]. They are therefore likely to be more similar than they are to more distant relatives. Hence, they allow investigators to make stronger inferences. From a statistical point of view, they are needed to transform the comparative data into data that can be applied to standard statistical analysis (e.g., regression analysis, ANOVA or chi-squared tests). That is, it must not violate the assumption that the data points are independent. In this approach, phylogenetic information needs to be incorporated because the evolutionary distance between the pairs must be considered when calculating its pairwise score.

We next describe in detail how the implemented algorithms that determine the divergence times and other phylogenetic characteristics work. In particular, we are interested in the last common ancestor node of any pair. This is useful for determining the distance between pairs of species in a tree, and this distance may affect the score.

Generating the phylogenetic tree The first phylogenetic-related algorithm is the generation of a tree structure from a NEWICK tree representation. Algorithms for that parsing procedure are publicly available[4] and therefore, a description is neglected. From now on, we assume that we have a tree-like data structure from the original NEWICK string.

Finding the last common ancestor node of two species We now present an algorithm as proposed by Gusfield [28, p. 194]. This algorithm reduces the last common ancestor problem to a list problem. More specifically, it reduces the general problem to a problem of finding the smallest number in an interval of a fixed list of numbers (the *range minima problem*), and the two main

[4] e.g., see the Mesquite source code for a java version, or the BioPerl library (www.bioperl.org/)

steps are as follows:

1. Preprocessing

   Execute a depth-first traversal of the tree to label the nodes in depth-first order. However, the only property of the numbering we need is that the number given to any node is smaller than the number given to any of its proper descendants. We then simply build a multi-list L of the nodes in the order they are visited.

2. Retrieval

   For a pair of nodes x and y, find any occurrences of x and y in L (that is, find the first position of either x and y, and their last position). This defines an interval I in L, and the smallest number in this interval is the last common ancestor of x and y.

This procedure can be verified as follows. For each last common ancestor retrieval (x and y, respectively), two distinct cases can be distinguished [34, p. 21]:

1. One node (say x) is an ancestor of the other (y): All those nodes visited between x and y are in the subtree of the ancestral node, and thus the depth-number assigned to x is minimal in I.

2. Neither x nor y is an ancestor of the other: All those nodes visited between x and y are in the subtree of last common ancestor (x, y), and the traversal must visit this ancestral node. Thus, the minimum of I is the depth-number assigned to last common ancestor (x, y).

The complexity of the algorithm is as follows. Suppose one has a tree with n leaves. Then, in the preprocessing, the time complexity is proportional to the number of vertices plus the number of edges, thus $\mathcal{O}(|V| + |E|)$, which is in $\mathcal{O}(n)$. The multilist allocates $\mathcal{O}(n)$ entries, as well as the depth-first traversal. Thus, space complexity is $\mathcal{O}(n)$. In the retrieval, $\mathcal{O}(n)$ time is needed to determine the smallest number in an interval. This can be further reduced to a constant retrieval time[2], however, it is not implemented in this application, because $\mathcal{O}(n)$ is also a acceptable boundary.

Additional phylogenetic information After calculating the last common ancestor node for a pair of species, the program retrieves information about branches and their length on the path that connects both species. This is done by using the following procedure, and is repeated for each pair of species:

1. For both of the two species nodes, traverse the tree from the tip to the last common ancestor node, and sum up the branch lengths; also count the number of branches that are included in this summation. Additionally, save the IDs of all visited edges and store them in the proper *PairwiseComparison* object.

2. The divergence time since the species split up is then just the sum of the branch lengths from all edges that are stored in this object.

This phylogenetic information is sufficient for both providing additional information about a pair-

wise comparison and for calculating the scores. In the next section, we provide more information on how this phylogenetic information is incorporated into the score calculation mechanisms.

4.3.5 Score Calculation

One problem that we face is that we need to define a way to evaluate the power of a pairwise comparison. That is, we want to systematically target pairwise comparisons that offer the most power to test hypotheses. However, a few issues can arise:

- The traits of interest have a very heterogeneous range. In general, continuous variables are more scattered and have therefore a larger range; they can also differ greatly in their range. In contrast, discrete variables have only a finite number of states. Usually, they have only two states (0 and 1) and are thus binary. But, nevertheless, all traits of interest should be weighted equal, so a transformation is needed.
- We need different scoring mechanisms for different kinds of evolutionary hypotheses.
- We have to define a score that comprises all of the trait information into a single value, representing how much power a specific pairwise comparison has.

The issue of the greatly heterogeneous variable ranges refers to all traits. It can thus be treated the same by finding a transformation that recodes the ranges to a common interval. We apply a linear scaling transform, because it is well-suited for this purpose and has the following properties [38, p. 4]:

- It introduces no distortion to the variable distribution.
- It has a one-to-one relationship between the original and normalized values.
- The variable range is always between 0 and 1 after the transform, independent of the original range.

In the application, the transform is done as follows. For each trait we do the following:

- Determine the minimum (min) and maximum (max) value of the variable / trait
- Use the following formula or a deviation of it to do the recoding:
  - Variant 1

$$y = \frac{x - \min}{\max - \min} \tag{4.2}$$

  - Variant 2

$$y = \frac{x - \max}{\min - \max} \tag{4.3}$$

- store the transformed value in an additional column (*adjusted score* S_{adj})

With this procedure and the introduction of *adjusted scores* for each trait, they all have exactly the same range $[0, 1]$ and are thus homogeneous. This is a good starting point to develop a sophisticated scoring system.

In the following, a detailed overview of the scoring system for different hypotheses is given. However, we describe the main hypothesis and alternative hypotheses scoring mechanism separately.

Scoring mechanisms for the main hypothesis

The scoring mechanism of the main hypothesis is straightforward: The bigger the difference, the higher the score. This is due to the fact that we are always interested in big differences in the main hypothesis, because this reflects evolutionary change. In detail, we apply the following procedure:

1. We calculate the difference between the two species.
2. If the difference is negative, we do a sign reversal for all traits (force the main hypothesis to be positive and guarantee the same direction of change for all pairwise comparisons). The reasoning for that is as follows: First, one wants to achieve consistency with other programs, such as CAIC [67] and PDAP-Mesquite [51]. Another argument concerns helping to make sense of the other trait differences and their directions. By making the main hypothesis always positive, it becomes possible to quickly see if other traits are consistently positively or negatively correlated with the trait in the main hypothesis, e.g. by simply sorting the columns for those traits and looking for a preponderance of positive or negative values.
3. We determine the maximum value in all considered pairs; the minimum value is always set to 0. This is desirable, because otherwise, non-zero values will be transformed to 0 after the linear transform, which would be confusing and unreasonable.
4. We apply the linear scaling transform as described above (using variant 1) and store this transformed value as the *adjusted score* for the main hypothesis.

Scoring mechanism for the alternative hypotheses

To enable the testing of different kinds of hypotheses (e.g., mutually exclusive and non-mutually exclusive), three distinct scoring mechanisms can be specified for each alternative hypothesis. For all these mechanism, the direction of change always refers to the direction of change in the main hypothesis. The three distinct scoring options are as follows, and they are also visualized in Figure 4.5:

Option 1: No change

Pairwise comparisons that make the change as small as possible are scored higher, whereas pairwise comparisons with big differences are scored lower. The variable range after applying this procedure is again between 0 and 1. It should therefore be applied to non-mutually exclusive hypotheses

when the effect of a potentially confounding variable should be included in the calculation. In practice, it works as follows:

1. The program determines the absolute values of the differences.
2. The program determines the maximum value; the minimum value is always set to 0.
3. The program applies the linear scaling transform as described above (using variant 2) and store this transformed value as the *adjusted score* for the particular alternative hypothesis.

Option 2: A lot of change in the opposite direction

Pairwise comparisons with big differences in the opposite direction as the difference in the main hypothesis are scored positively, whereas differences in the same direction are scored negatively. The variable range after applying this procedure is now between -1 and 1. This scoring scheme is useful for hypotheses that are mutually exclusive, but might be problematic when they are not: The positive effects of one variable and the negative effects of the other might 'wash out' any effect. In practice, it works as follows:

1. The program splits the trait array in two parts, one part contains only all non-negative values, the other part all negative ones.
2. In each part, the program determines the minimum and maximum values separately. In the array that contains all negative values, the maximum is always set to 0, whereas that is the case with the minimum in the array that contains all non-negative values.
3. The program applies the linear scaling transform as described above (using variant 1) on both arrays separately and merge the arrays back together.
4. The program applies a sign reversal for the array (because changes in the opposite direction should receive a positive score, and changes in the same direction should receive a negative score).
5. The program stores the array as the *adjusted score* for this hypothesis.

Option 3: A lot of change in the same direction

Pairwise comparisons with big differences in the same direction as the difference in the main hypothesis are scored positively, whereas differences in the opposite direction are scored negatively. The variable range after applying this procedure is again between -1 and 1. This scoring scheme is the exact opposite of the one stated above, and it is also useful for mutually exclusive hypotheses. However, in this case, when positive increases in two independent variables result in different effects for the dependent variable, we have to score one effect higher. For example, if an increase in the independent variable reduces the dependent variable in one pairwise comparison, but increase in the independent variable also increases the dependent variable in another pairwise comparison, one might want to score differences in the same direction more highly.

In practice, it works exactly as option 2 with the only exception that the programs does not apply a sign reversal after the linear scaling transform.

Summary

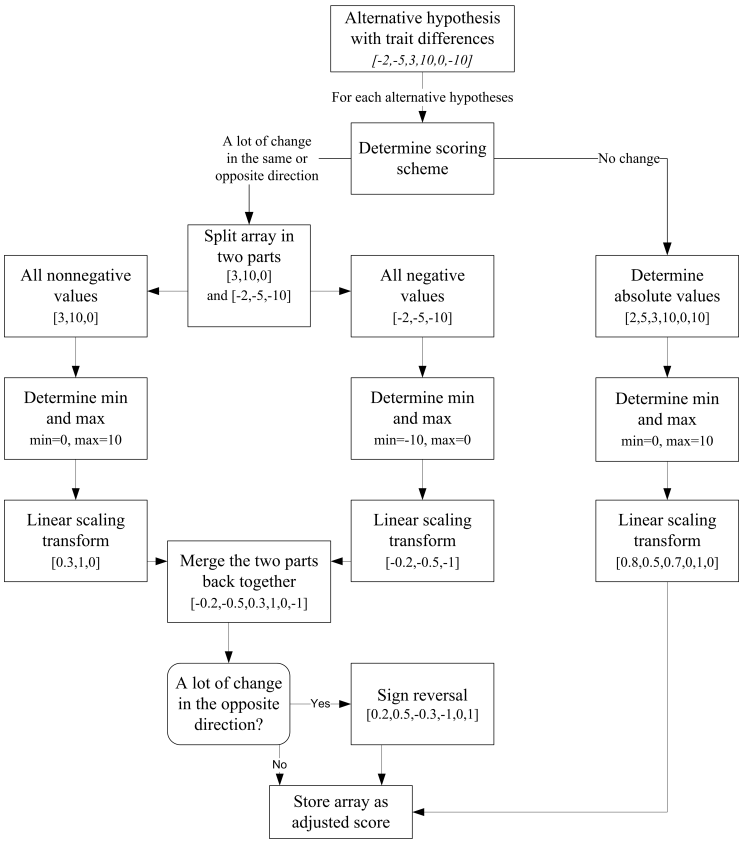


FIGURE 4.5: Overview of all scoring schemes for alternative hypotheses with a concrete example: An array with trait differences is applied to the scoring scheme. See also text for details.

In summary, all traits that represent a hypothesis are weighted equally; they all play equal roles in assessing which species offer the strongest tests of a hypothesis. Furthermore, only the specified traits from the setting step are considered in the calculation. Alternative hypotheses can have a negative influence on the raw score, reflecting that it is contradictory to the expected outcome regarding the hypotheses and scoring procedure. Nevertheless, a pairwise comparison can still be informative, even if one trait is contributing a negative score. This is a major advantage of the whole approach, since it does not completely eliminate pairwise comparisons that are partially conflicting to the hypotheses. Instead, those pairs are 'punished', but still included. If a user wishes to have values that are always opposite or the same as the main hypothesis, it is easy to pick these out, due to the fact that that the main hypothesis is positivized.

Final score mechanisms

The number of scores for a particular pairwise comparison must be reduced to one to provide and facilitate the evaluation of different pairwise comparisons on the tree. Therefore, new variables have to be introduced that reflect all the information from the trait scores, as well as phylogenetic information, to establish a basis to compare among pairwise comparisons with different evolutionary distances.

Unstandardized raw score

All *adjusted trait scores* are summed up to define the unstandardized raw score for a pairwise comparison,

$$S_{raw\_u} = \sum S_{adj} \tag{4.4}$$

, where S_{adj} denotes an adjusted score for one of the traits of interest, as described earlier. It comprises all the information into a single value and thus, it represents how compelling they are to test the specified hypotheses. The range of this score is $[-1 * N_{A_D}, 1 + N_A]$, where N_A is the number of alternative hypotheses and N_{A_D} the number of alternative hypotheses with a score mechanism unequal to 'no change'. The reasoning is as follows: The scoring scheme of the obligatory main hypothesis has a range of $[0, 1]$, as well as the first scoring scheme of alternative hypotheses (no change). Only the second and the third scoring scheme of alternative hypotheses (a lot of difference in either of the two directions) have a variable range of $[-1, 1]$. Thus, the asymmetry arises because the number of alternative hypotheses can be arbitrary, and different scoring schemes have different ranges.

Standardized raw score

However, this defined raw score can sometimes be uninformative when compared to different pairwise comparisons. This is due to the fact that in general, the more divergent two species are, the more likely it is that they evolved bigger differences. Therefore, it is natural that distant species pairs have higher differences than congeneric species pairs; an equivalent formulation is that different pairs have a different variance. The method of pairwise comparisons has not traditionally controlled for this and the standardization is basically a heuristic approach to help identify further species to study, rather than an explicit analysis based on correlation, regression or other parametric tests (where standardization is more important).

In our approach, the program overcomes the problem of different variances by introducing another variable, the standardized raw score, which is simply the normalization by the standard deviation. For an arbitrary pairwise comparison (x, y), it is defined as

$$S_{raw\_s} = \frac{S_{raw\_u}}{\sqrt{\sum_{e \in (x,y)} w_e}} \tag{4.5}$$

, where e denotes an edge on the path from x to y, and w_e the weight (or length) for edge e. Thus, the unstandardized raw score is transformed by dividing it by the square root of the sum of the branch lengths, and the normalization transforms the trait differences in a way that all pairs behave like they have a total branch length of 1. Thus, all pairwise comparisons have a common variance as required by most statistical tests and different raw scores can be compared, regardless of the evolutionary time since they last shared a common ancestor. Another argument for controlling for branch lengths is that fewer traits should change on shorter branches, and thus it helps control for confounding variables. Despite its advantages, the standardization can be turned

off in step 2 if desired. This can be also useful, because the method of pairwise comparisons does not have an underlying evolutionary model and standardization might be not needed, because we sometimes expect a larger absolute change in some trait, regardless of its rate of change, to be more valuable than a small change over a short branch. For example, brain size that increases by an order of magnitude might be a stronger test than a smaller amount of brain change, even if it occurs over a small branch. The choice is up to the user, based on their knowledge of the system and hypotheses.

If the standardization is disabled, then both raw scores are equal. However, it is the deciding score in the whole scoring system, and it is the score that is considered when choosing among pairs of species that offer the most power to test hypotheses.

Pairing score

The pairing score ('pairing' as suggested by Maddison [44]) reflects the score of a whole set of pairs (see section 4.3.8 for a definition). It is the sum of all selected pairwise comparison scores and represents how compelling the set of species is to test the hypotheses, given the user-specified settings related to the compatibility of the hypotheses. In the application, the pairing score is displayed above the summary table, whenever at least one pair is selected.

Factors that influence the raw score

The scoring system is very crucial since it forms the basis for the decision on which pairwise comparisons offer the most power. Thus, it is also important to be aware of the main factors that influence these scores. Most of the reasons are evolutionary based; however, less obvious factors also contribute to the score and are therefore worth noting. The following list describes the most important factors that influence the raw score of a pairwise comparison:

- **Clear-cut difference in the main hypothesis**: Pairwise comparisons with a big difference in the main hypotheses indicate that there has been more evolutionary change. This leads to higher power, due to a stronger effect size.
- **Control for confounds and alternative hypotheses**: Depending on the particular question, it may be reasonable to control for confounding variables or investigate alternative hypotheses. This can be controlled by specifying alternative hypotheses, and they can also contribute to the power of a pairwise comparison by 'favoring' only appropriate pairwise comparisons, according to the chosen settings. Each alternative hypothesis changes the final *raw score*, because it is the sum of all hypotheses traits.
- **Differences that happened in a short period of time**: It is generally interesting if big differences in a certain trait happened in a small amount of time. This could potentially indicate evolutionary pressure, and should be rewarded with more power.
- **Correlated predictor variables**: Another factor to consider is whether the predictor variables are correlated. For mutually exclusive hypotheses we expect that if two traits are uncorrelated, there will be more contrasts where the first is one sign, and the second is the opposite. However, if they are highly correlated, we will have less power to distinguish among them, because we would have fewer contrasts that differ strongly. Similar logic could apply

to non-mutually exclusive hypotheses – it will be more difficult to find cases where change in one trait has little change in the other trait, and so correlation among the predictor variables will reduce the power to test compatible hypotheses. Nevertheless, this all depends on the direction of correlation and the specific predictions.

- **Selection of species**: It also depends on the selection of species one is looking at: If the species diversity is high, than the differences are likely to be big in some pairs, because a large amount of time is present since these species split up. Thus, small differences achieve only inconsiderable power, because much greater differences in more diverse species pairs are present. Otherwise, if one is investigating only close relatives, than the differences are probably much smaller, but, nevertheless, informative. Therefore, we have to give these small, but still extremal, differences more power, because they are the most compelling in the considered subset of species.

4.3.6 Target Variable

In step 2, the user can choose to specify a target variable. The purpose of this feature is twofold: First, it provides a mechanism that easily selects those pairs of species that have already been studied or not studied in relation to a specific dependent variable, such as sleep durations or a cognitive task. We might want to make use of this information to identify other species that should be studied compared to these already studied species. Secondly, it enables the investigator to see the distribution of the studied species in the tree (see Figure 4.6).

A target variable must be a discrete binary variable, because the nature of this variable questions if data are available or not for a particular species[5]. From all traits, *PhyloTargeting* automatically detects those that come in question to be a target variable. It should also be noted that the target variable does not influence the pairing score; instead, it is just an additional selection factor that further screens pairings for their relevance to the hypotheses of interest.

We now describe both purposes in more detail:

- The first purpose is a screening measure among all possible pairs. It can be seen as an additional constraint that must be fulfilled if a particular pair should be considered in the analysis. After a target variable is specified, different options for this selection process can be specified:
  - Data are available for both species (marked as '2')
  - Data are available for one species (marked as '1')
  - Data are available for at least one species (marked as '1' or '2')
  - Data are available for none of the species (marked as '0')

  These options are intuitive: If option 1 is selected, only those pairs are considered that have no missing data in both species that form the pair; all other pairs are neglected. For example, if the target variable is a variable where data are available for all species, no pair will be neglected. However, if only a small amount of species have been studied in relation to this variable, most of the pairs will be neglected and only those which contain these studied species are left. This concept can be applied to the other options as well.

[5] data available that is coded as 1 / no data available that is coded as 0

Species		Target var. (CognitiveStudy)
Species 1	Species 2	# Species where data are available
Eulemur_macaco	Eulemur_rubriventer	0
Eulemur_rubriventer	Lemur_catta*	1
Eulemur_macaco	Eulemur_mongoz	0
Eulemur_mongoz	Lemur_catta*	1
Hapalemur_aureus	Lemur_catta*	1
Lemur_catta*	Lepilemur_mustelinus	1
Hapalemur_griseus*	Lemur_catta*	2
Daubentonia_madagascariensis*	Lemur_catta*	2
Cheirogaleus_major	Lemur_catta*	1
Eulemur_coronatus	Lemur_catta*	1
Eulemur_fulvus*	Eulemur_rubriventer	1
Avahi_laniger*	Lemur_catta*	2
Lemur_catta*	Varecia_variegata	1
Eulemur_fulvus*	Eulemur_mongoz	1
Lemur_catta*	Microcebus_coquereli	1
Lemur_catta*	Phaner_furcifer	1

Species		Target var. (CognitiveStudy)
Species 1	Species 2	# Species where data are available
Eulemur_macaco	Eulemur_rubriventer	0
Eulemur_rubriventer	Lemur_catta*	1
Eulemur_macaco	Eulemur_mongoz	0
Eulemur_mongoz	Lemur_catta*	1
Hapalemur_aureus	Lemur_catta*	1
Lemur_catta*	Lepilemur_mustelinus	1
Hapalemur_griseus*	Lemur_catta*	2
Daubentonia_madagascariensis*	Lemur_catta*	2
Cheirogaleus_major	Lemur_catta*	1
Eulemur_coronatus	Lemur_catta*	1
Eulemur_fulvus*	Eulemur_rubriventer	1
Avahi_laniger*	Lemur_catta*	2
Lemur_catta*	Varecia_variegata	1
Eulemur_fulvus*	Eulemur_mongoz	1
Lemur_catta*	Microcebus_coquereli	1
Lemur_catta*	Phaner_furcifer	1

FIGURE 4.6: An example screenshot that illustrates the function of the target variable (3rd column). On the left side, no target variable constraint is specified, and all pairs are displayed, independent of the value in the target variable. However, on the right side, after specifying that only pairs should be considered that have exactly one missing value in either of the two species in the target variable (marked as '1'), all pairwise comparisons where data are available for either 0 (marked as '0') or both (marked as '2') species would be removed from the set and are marked gray.

- It allows to quickly 'pinpoint' where the missing data points are, which helps identifying potential biases as discussed in Chapter 1. The implementation is simple, because the program must only store a binary variable for every species, i.e. according to whether it has been studied. The graphical visualization of this issue is described in section 4.4.3.

4.3.7 Contrast Selection

To test hypotheses in a comparative context or, as in this approach, to test which species should be studied regarding particular hypotheses, species values needs to be compared and their differences (contrasts) must be calculated. The more contrasts we have, the more likely it is that patterns are detected. Moreover, with a growing number of contrasts, a higher proportion of the true phylogenetic diversity of a lineage is considered. Different contrasts must be phylogenetically separate, and this criterion is of utmost importance. Hence, we need a feature that enables us to select species pairs along the tree (and thus creating a pairing that consists of more and more pairs), while guaranteeing that the phylogenetically separate constraint is always fulfilled. This unique feature is the *contrast selection* procedure and it allows the selection of a pairing in real-time and automatically guarantees that it is phylogenetically separate. That is, no branch is used twice, and thus, statistical independence is assured. Generally speaking, it also allows the investigator to visually see the dependencies among the pairwise comparisons along the tree. In practice, it is intuitive and fast, and allows a real-time investigation of all pairwise comparisons. Internally, it works as follows:

- The investigator selects a pair, for whatever reason (e.g., the pair has either a high score, or

it is informative in another way).

- The algorithm automatically determines all pairs that share at least one branch with the just selected pair and inactivates these pairs by 'locking' them (see Figure 4.7).
- The investigator can select more pairs, until no pair can be selected due to the phylogenetically separate constraint. The investigator can now either 'accept' this pairing or change it by deselecting already selected pairs.
- The pairing score is determined (sum of the raw scores of all selected pairs).

Species 1	Species 2	Contrasts (Unset)
Eulemur_macaco	Eulemur_mongoz	Choose
Eulemur_macaco	Eulemur_rubriventer	Choose
Otolemur_crassicaudatus	Otolemur_garnettii	Choose
Eulemur_fulvus	Eulemur_rubriventer	Choose
Eulemur_rubriventer	Lemur_catta	Choose
Eulemur_fulvus	Eulemur_mongoz	Choose
Eulemur_mongoz	Lemur_catta	Choose
Hapalemur_griseus	Lemur_catta	Choose

Species 1	Species 2	Contrasts (Unset)
Eulemur_macaco	Eulemur_mongoz	LOCKED
Eulemur_macaco	Eulemur_rubriventer	LOCKED
Otolemur_crassicaudatus	Otolemur_garnettii	Choose
Eulemur_fulvus	Eulemur_rubriventer	1.ACTIVE (DESELECT)
Eulemur_rubriventer	Lemur_catta	LOCKED
Eulemur_fulvus	Eulemur_mongoz	LOCKED
Eulemur_mongoz	Lemur_catta	LOCKED
Hapalemur_griseus	Lemur_catta	Choose

FIGURE 4.7: Example of the *contrast selection* procedure: At the beginning, no pair is selected (left). However, after selecting the *Eulemur fulvus* - *Eulemur rubriventer* pair, all phylogenetically non-separate pairs are automatically determined (right).

From an informatics point of view, this is implemented as follows:

- Every pair stores the IDs of the allocated branches in an associative array ('branch array').
- At the beginning, every pair is selectable.
- Whenever a pair is selected, all phylogenetically non-separate pairs (regarding the set of selected pairwise comparisons) are determined[6]. These pairs are then removed from the set of selectable pairs and marked. The selected pair changes its state to *selected* (see section 4.3.2), and this process continues unless no pair is selectable anymore.
- The pairing score is determined by summing over the *raw score* of all selected pairs.
- Whenever a pair is deselected, all dependencies as well as the pairing score are recalculated.

In summary, a pairwise comparison can have three different states in the rightmost column in the summary table:

- *SELECT*: This pairwise comparison can still be selected. By clicking on the 'SELECT' link, the particular pairwise comparison changes its state to *selected* and the row is marked blue. Finally, the algorithm described above determines all phylogenetically non-separate pairs.
- *LOCKED*: This pairwise comparison cannot be selected anymore, because at least one already selected pairwise comparison shares at least one common branch. These pairs would then be phylogenetically non-separate and thus, no selection is possible.
- *SELECTED*: This pairwise comparison is already selected (indicated by a blue row). That is, the path connecting the two species is allocated in the tree, and no other path may cross.

[6] that is, the intersection of the branch arrays is not empty.

By clicking on the 'DESELECT' link, the selection is removed and the connecting path is freed. To reset all selected pairwise comparisons immediately, the user can also click on 'Unset' in the rightmost table header column.

4.3.8 Maximal pairing

The previous section introduced a way to generate a pairing. This is a crucial task and one immediate apparent criterion is to select a pairing that gives the highest overall score, as this would indicate the set of species that gives the most power to test the hypotheses. However, with large data sets and a high number of pairwise comparisons that needs to be considered, this is difficult to achieve manually, due to the complex nature of the pairs that must not share a branch. Hence, we developed an algorithm that automatically determines the pairing with the highest overall score, which we call the maximal pairing.

Definitions

We need some terms and definitions for the following sections. To our knowledge, some terms were first introduced by Maddison [44], and we use a similar terminology to avoid confusion. This may also help to establish these terms in the scientific community.

Definition (degree of a node): *The degree of a particular node is defined as the number of children.*

Definition (path between two taxa): *The path between two taxa is the path on the tree that links the two taxa. That is, we begin at the ancestral node (the last common ancestor of the two taxa) and proceed tipward in two directions, on each reaching a different terminal taxon.*

Definition (pair): *A pair is a set of two terminal taxa as well as their corresponding path.*

These three definitions are straightforward and illustrated in Figure 4.8.

Definition (Phylogenetically separate pair, PSP): *A pair of terminal taxa is denoted as phylogenetically separate*[7] *if none of the paths from other pairs touch or cross their path.*

In literature, this is often called *phylogenetically independent*. However, as Maddison mentioned, this is misleading because the pairs might be not independent in a statistical sense. Hence, the term *phylogenetically separate* is preferred. Two pairs are phylogenetically separate if and only if they do not share a common branch. This idea is crucial, and illustrated in Figure 4.8.

Definition (v-rooted PSP or v-rooted pair): *Given an arbitrary internal node v, a particular PSP is denoted as v-rooted PSP or v-rooted pair w.r.t. v if v is the last common ancestor node of the PSP.*

This definition is new and we need it for the maximal pairing algorithm to explain the basic concepts.

Definition (pairing): *A pairing is a set of PSPs on a tree.*

[7] A phylogenetically separate pair and a pairwise comparison represent the same entity throughout this thesis. However, if we want to highlight the phylogenetically separate constraint, we prefer the term PSP, whereas we prefer the term pairwise comparison (sometimes also pair for simplicity) if this constraint is not important in the specific context.

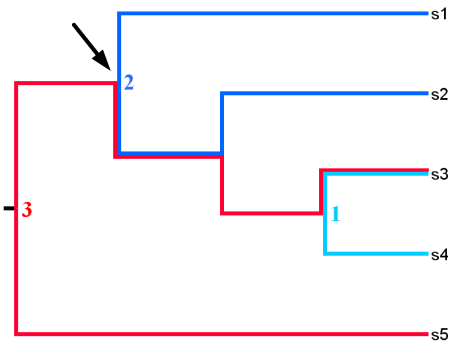


FIGURE 4.8: An exemplary sample tree for the terms introduced in this section. Three pairs (1, 2, and 3) and four interior nodes (all have degree 2) are illustrated. With respect to the arrow-marked node, pair 2 is a rooted PSP, whereas pair 1 and pair 3 are not. Furthermore, pairs 1 and 3 are not phylogenetically separate, whereas 1 and 2 are. Thus, the set of the three pairs is no pairing, because two branches are used twice. See also text for details.

As one can see from the definition, pairings must have the property that they are phylogenetically separate. That is, the evolutionary paths linking the terminal taxa cannot be shared. In general, more than one pairing exists that satisfies the *phylogenetically separate* criterion and we have to choose among different pairings. This automatically leads to the following definition:

Definition (maximal pairing): *A maximal pairing is a pairing that maximizes the score of the pairing. That is, the pairing score is maximal among all possible pairings.*

Definition (score of the maximal pairing): *The score of the maximal pairing is the sum of all PSP scores that belong to the maximal pairing.*

The so-defined maximal pairing is the pairing that comprises the most power to test the specified hypotheses. It is therefore a fundamental concept of this work, and will be explained in great detail in this section. Figure 4.9 shows all possible pairings for a small example tree.

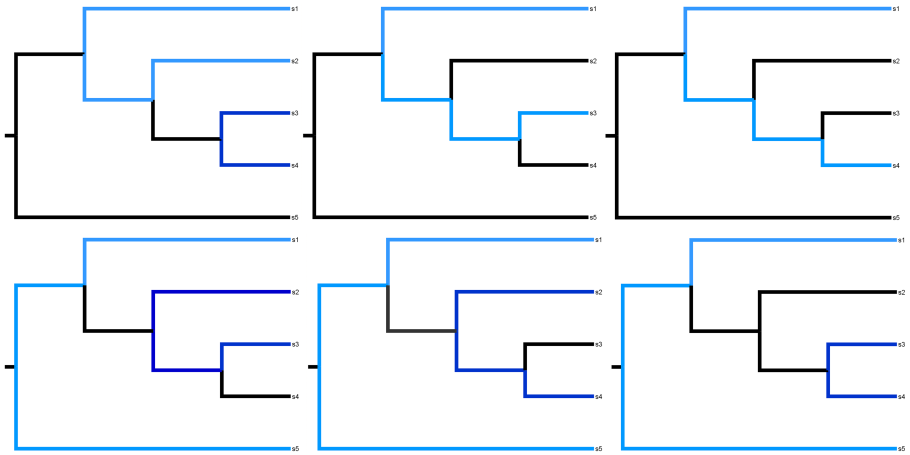


FIGURE 4.9: All six possible pairings for an example tree consisting of five species. Which one should be preferred?

Definition (availability-code): *The availability-code for a node v is a string of length degree(v) consisting of characters 0 and 1, where character 1 at position i indicates that the edge that connects v and the i-th child is not allocated by a PSP, whereas 0 indicates the opposite. We call an availability-code smaller compared to another one if the number of character 0 is bigger.*

This concept is fundamental for polytomous nodes, and explained throughout this section (see also Figure 4.10).

Preliminary considerations

At first, we briefly discuss some concepts that are needed to calculate the maximal pairing.

1. **Preprocessing**
   We have to do some preprocessing steps before the maximal pairing algorithm is called. This is due to the fact that we need additional information in the algorithm and it is much more efficient to compute this information initially rather than repeatedly while the algorithm runs. These measures greatly improve the execution time, an issue that is further discussed in Chapter 5.
   Specifically, we have to determine all pairs that belong to a particular node. This procedure, hereafter called *pair allocation*, is straightforward and described only briefly. For every pair ($\frac{n(n-1)}{2}$ at most), we have to determine the ancestral node x that links both paths. This node represents the last common ancestor of the PSP and the PSP is thus an x-rooted PSP. We then store the ID of the PSP in node x. This procedure allows a $\mathcal{O}(1)$ lookup for PSPs that belong to a particular node, an issue that greatly reduces the execution time of the algorithm, especially for polytomous nodes.
2. **Determine all leftover subtrees for a particular PSP**
   This procedure, hereafter called *subtree identification*, is called whenever a PSP has been selected. The purpose is to determine all leftover subtrees that arise with this particular PSP (see also Figure 4.10). Subtrees of size 1 (that is, terminal nodes) are neglected, because no PSP can be selected in that subtree if only one node is available. If the root of a subtree is a polytomous node, we also have to retrieve information about the availability-code. This concept guarantees that the maximal pairing is indeed found, and it incorporates the special property of polytomous nodes that more than one rooted PSP can be selected without violating the phylogenetically separate constraint.

General Diagnosis

A brute-force approach to determine the maximal pairing is unsatisfactory, because this is in general not feasible since the number of possibilities grows exponentially with the number of species. Therefore, an algorithm is needed that will eliminate most of the possibilities due to certain constraints and thus makes computation of the maximal pairing feasible. Indeed, a dynamic programming algorithm can be developed, based on the following idea: Imagine the maximal pairing on a tree. There are two distinct possibilities for the root node, hereafter called r:

1. **No PSP goes through the root of the tree**

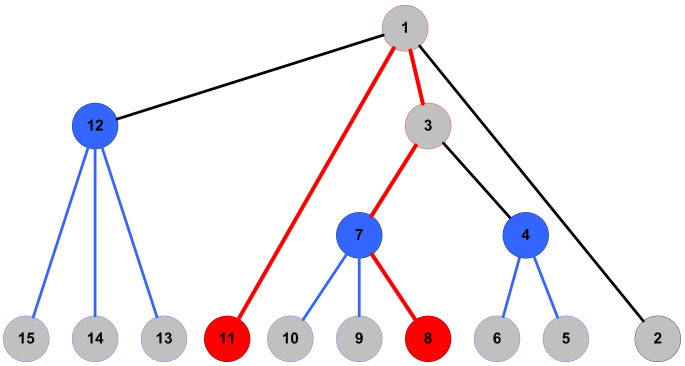


FIGURE 4.10: Illustration of the leftover subtrees and availability-code concepts when a PSP is selected. In this example, a rooted PSP (8 versus 11) for node 1 is shown, which yields to three non-trivial subtrees that have to be evaluated: Two of them are polytomous (7 and 12); one of them is dichotomous (4). For each polytomous subtree, not only the node itself must be stored, instead we also need information about the availability-code. As highlighted in the example tree, all children of node 12 are 'free' ('111') when pair 8-11 is selected, whereas only node 10 and node 9 are 'free' for node 7 ('110', left to right). For dichotomous nodes, as with node 3, this information is redundant.

In this case, hereafter called *case 1*, no r-rooted PSP exists, and all PSPs that belong to the maximal pairing must be located in the descendent subtrees (children). Thus, the problem can then be decomposed into smaller instances, because subtrees do not share any branches and can thus be treated separately. For example, if r has a degree of four, then we decompose the problem into four smaller instances and recursively call each children separately.

2. **At least one PSP goes through the root of the tree**

   In this case, hereafter called *case 2*, at least one r-rooted PSP is selected, and we have to call the *subtree identification* procedure for each selected r-rooted PSP. Each leftover subtree can be evaluated separately, using recursive calls.

Both cases allow a decomposition of the initial problem into smaller instances, which is essentially the basic idea of the dynamic programming approach. It can be defined as a general design technique that exhibits the property of overlapping subproblems. By solving smaller instances once, recording the solutions in a table and finally extracting these solutions to the initial instance from the table, we can design efficient algorithms.

Maximal pairing algorithm

In practice, however, we do not know which set of PSPs yield to the maximal pairing. Thus, we have to develop an algorithm that proceeds from the root of the tree up to the leaves (top-down approach) to determine the maximal pairing. Assuming that we want to determine the score S of a particular node T, the algorithm works as follows:

1. **Determine the score of case 1**

   This includes recursive calls for all children and summing up their scores. If a particular child is polytomous, we call it with an availability-code consisting of a series of the character

1, indicating that we have to consider all children. Specifically, we have

$$S_{T_{\text{case 1}}} = \sum S_{\text{desc}(T)} \tag{4.6}$$

, where desc(T) represents all children of T.

2. **Determine the score of case 2**

   This includes the determination of at least one T-rooted PSP that maximizes the following term:

$$S_{T_{\text{case 2}}} = \max_{R}(S_R + \sum S_{\text{subtrees(R)}}) \tag{4.7}$$

   where R is a rooted PSP for node T, S_R the score of R and *subtrees(R)* represents all leftover subtrees in relation to R.

   If the node is dichotomous, we have to determine the maximum of all possible T-rooted PSPs and their arising subtrees. If the node is polytomous, however, this technique is not sufficient, because more than one T-rooted-PSP can be chosen, as mentioned earlier. Instead, we have to determine the maximum for all possible combinations of T-rooted PSPs and their arising subtrees. Thus, polytomous nodes are recursively called multiple times, until no more rooted PSP can be selected, and every call decreases the availability-code.

3. **Determine the maximum of both cases**

   Finally, we compare these two cases, and the maximum of both is the actual score of the node. If case 2 scored more positively, than we also have to store the IDs of all selected PSPs in the node object, which is necessary for the backtracking procedure.

Graphically, we can illustrate the decomposition of the two cases as follows (shown for a dichotomous tree):

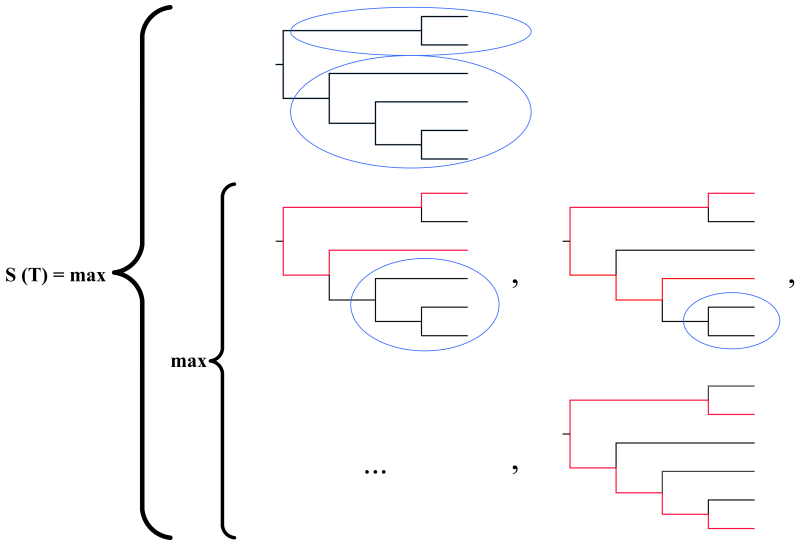


FIGURE 4.11: Graphical representation of the decomposition. See text for details.

We immediately arrive at the following recursion that calculates the score S_T for a particular node T:

$$S_T = \max \begin{cases} \sum S_{\text{desc(T)}} \\ \max_R \ (S_R + \sum S_{\text{subtrees(R)}}) \end{cases} \tag{4.8}$$

with initial conditions $S_T = 0$ if T is a leaf. Note that in this formula, in the case of polytomous nodes, a node can be a subtree of itself, but with a smaller availability-code.

Backtracking

After computing the score of the maximal pairing, we have to reconstruct the PSPs that belong to the maximal pairing. This procedure is typical for most dynamic programming algorithms: One first calculates the value of the optimum and one then reconstructs the solution, based on the information collected in the forward recursions. Again, we apply a top-down approach to proceed the tree from the root to the leaves to reconstruct the maximal pairing.

For a particular node n, we first determine which of the two cases achieved the higher score. If case 1 scored higher, then we recursively check all children with the same procedure. If one of these subtrees is polytomous, then we call this particular subtree with an availability-code that consists of a series of the character '1', representing that all children have to be incorporated. If case 2 scored higher, however, we have to distinguish between dichotomous and polytomous nodes.

Dichotomous nodes For dichotomous nodes, this is remarkably easy, because only exactly one n-rooted-PSP goes through the node. This n-rooted-PSP is uniquely determined[8], and we simply add its ID to the maximal pairing. We then have to determine all leftover subtrees from this particular PSP. The backtracking procedure is then called recursively for every leftover subtree. If one of these subtrees is polytomous, we call it with a particular availability-code (which contains also 0-characters in this case), as described earlier.

Polytomous nodes For polytomous nodes, the backtracking procedure is more complex, because more than one n-rooted PSP can be selected without violating the phylogenetically separate constraint. We apply the following algorithm to determine all PSPs that belong to the maximal pairing:

1. Call the polytomous node with a particular availability-code. Perform a lookup if a rooted PSP belongs to that availability-code.
   - If not, no more rooted PSPs belong to that node in the maximal pairing. We recursively call the backtracking procedure for all children where the availability-code is 1, and abort. This ensures that we also incorporate the scores of all 'free' children.
   - If yes, at least one more rooted PSP belongs to that node in the maximal pairing. Go to step 2.
2. Add the rooted PSP that is associated with the availability-code to the maximal pairing, and recursively call the backtracking procedure with all leftover subtrees from that rooted

[8] based on the collected information in the forward recursions

PSP. Update the availability-code with a new smaller one that reflects the allocation of the path from the last added rooted PSP. Go to step 1.

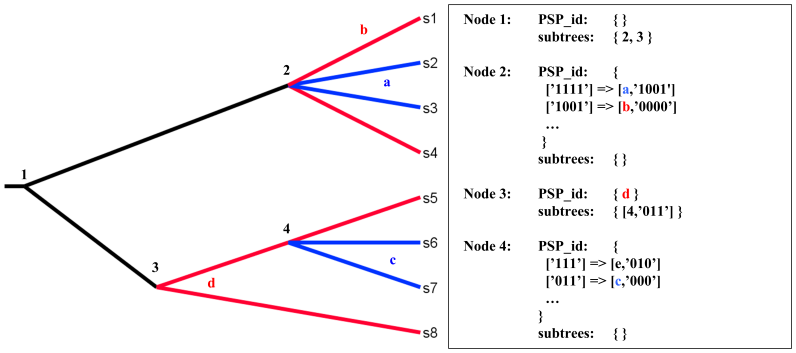


FIGURE 4.12: Backtracking Overview. At the left, a phylogeny with the maximal pairing is shown. At the right, parts of the backtracking concepts are presented; see also text for more details. For simplicity, only interior nodes are labeled and only relevant parts of the arrays are shown. PSP e ($s5$ - $s7$) is not highlighted in the tree. It does not belong to the maximal pairing of the tree, because PSP d already allocates the edge to $s5$. However, it does belong to the maximal pairing when the subtree rooted at node 4 is considered.

CPU and memory complexity

1. **Preprocessing** The pair allocation procedure needs $\mathcal{O}(n^2)$ time and space, because the ID of each PSP is assigned to exactly one node. The allocated space is deleted afterwards.
2. **Maximal pairing algorithm**
   - **Dichotomous trees** The recursion formula can be also seen as follows: For every possible path between two tips ($\mathcal{O}(n^2)$ in total), we have to evaluate all leftover non-trivial subtrees. The complexity to determine these subtrees is proportional to the length of the path. Thus, $\mathcal{O}(n)$ time is needed to determine all leftover subtrees for a particular pair, yielding to a $\mathcal{O}(n^3)$ algorithm in total. This is especially true for pectinate trees; for balanced trees, however, only $\mathcal{O}(\log_2 n)$ time is needed to determine the leftover subtrees, because the height of the tree is only $\log n$. Thus, for balanced trees, the maximal pairing can be computed in $\mathcal{O}(n^2 \log_2 n)$ time. Furthermore, $\mathcal{O}(n)$ space is needed to store the scores.
   - **Polytomous nodes** The score for a polytomous interior node equals the dichotomous case, plus an additional factor $2^{deg(node)-2}$ that accounts for all combinations of calls with a particular availability-code.
3. **Backtracking** Backtracking can be computed in $\mathcal{O}(n^2)$ time, because $\mathcal{O}(n)$ nodes have to be evaluated, and for each node, $\mathcal{O}(n)$ time is needed to identify the subtrees if case 2 scored higher. In practice, however, the procedure will be much faster due to several reasons:
   - Only $\mathcal{O}(1)$ time is needed if case 1 scored higher.
   - The number of nodes that have to be evaluated decreases if case 2 scored higher.
   - Only $\mathcal{O}(\log_2 n)$ time is needed for balanced trees to identify the subtrees.

Improvements

The execution time of polytomous trees can be improved to a polynomial algorithm by solving a maximal weighted matching problem for each polytomous node. This can be done as follows: Suppose the path of a particular PSP (a,b) goes through a polytomous node v. Let w be the child of v that is allocated by PSP (a,b). Then we need to compute the score of the leftover subtree $T(v)\backslash T(w)$. To this end we need to optimize over all v-rooted PSPs (except the v-rooted PSPs that go through w) and 'singleton' children u of v. The former represents case 2, and the latter case 1 in the maximal pairing.

This can be viewed as a matching problem and the following auxiliary graph G. The vertex set consists of all the children u of v except w, and a twin u^* for each u. We insert an edge in G for each pair of children u, u' and between u and its twin u^*. The weight of an edge (u, u') is the score for the v-rooted PSP, including the scores of all leftover subtrees. The score for a 'twin edge' (u, u^*) is the score of case 1 in the maximal pairing of node u. The optimal score that can be obtained on $T(v)\backslash T(w)$ is then the maximum weight of a matching in G, which can be computed in $\mathcal{O}(2(deg(v)-1)^3)$, using the algorithm of Gabow [21].

Concluding remarks

At the end, the following reasons are possible why species or species pairs have not been selected:

- A particular species has been excluded from the analysis (which is indicated in the graphical representation)
- A particular species has missing data relevant to either of the hypotheses, which also results in exclusion of the species
- A particular pairwise comparison does not fulfill the target variable constraint
- The raw score of a particular pairwise comparison is 0
- No free path to other species exist
- Other pairwise comparisons have higher scores and have therefore been preferred

Furthermore, we want to point out that the maximal pairing does not necessarily select the pairwise comparisons that have the highest scores of all considered pairs, because those high-scoring pairwise comparisons usually exclude too many other pairs in in order to ensure the phylogenetically separate constraint. However, it is guaranteed that no other set of pairs yields to a higher overall (pairing) score.

4.4 Analysis and Visualization

In this section, we describe analysis and visualization measures that have been developed to facilitate the interpretation and the general understanding of the results. To give an initial overview, we refer to Figure 4.13, which graphically presents some of the methods.

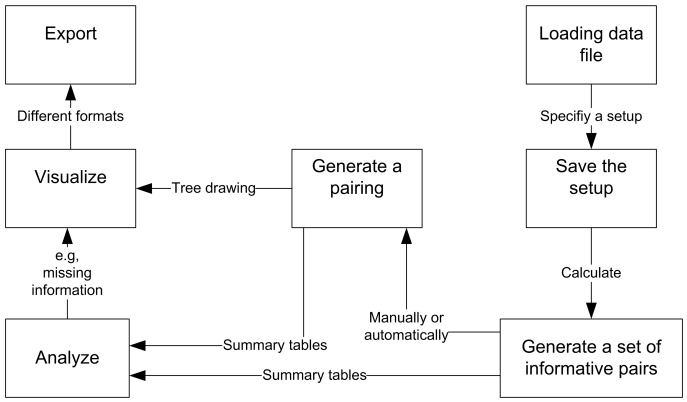


FIGURE 4.13: Overview of the analyses and visualization options.

4.4.1 Summary Tables

One main purpose of the program is to automatically calculate all possible pairwise comparisons. Since this number can become very large rapidly, it is indispensable to display them in a readable modality. We therefore provide summary tables of different types. They facilitate the search for a particular pairwise comparison, the analysis of it, or simply the export to a generally known file format. They have the following properties:

- All informative pairwise comparisons are shown.
- For each of them, information about the following properties are provided:
  - Scores for all traits (for each trait, raw score and adjusted score are displayed)
  - Phylogenetic information (the sum of the branch lengths and the number of included branches)
  - Target variable (if specified)
  - Final scores (the unstandardized and the decisive standardized raw score)
  - The pairing score, if at least one pairwise comparison has been selected
- The possibility to select particular pairwise comparisons via *contrast selection* (only available in step 3)
- The specified pairing can be graphically illustrated (see next section).

Summary tables can be found in different locations, and each of them has a distinct meaning:

- **Show all informative pairwise comparisons**: All informative pairwise comparisons are shown (step 3).
- **Show only pairwise comparisons from one species**: One can choose a species and display only comparisons from this particular species (accessible from step 4). This makes it easier to investigate pairwise comparisons from one specific species.
- **Show only selected pairwise comparisons**: With this option, only pairwise comparisons that belong to the specified pairing are displayed (accessible from step 4). This pairing can then be further analyzed and graphically displayed.

Species		Main hypothesis (Group_Size, continuous)		Phylogeny		Scoring		Choose contrasts
Species 1	Species 2	Raw difference	Adjusted difference	Sum of branch lengths	# of included branches	Sum of adjusted scores	Standardized raw score	Contrasts [Unset]
Allocebus_trichotis	Arctocebus_calabarensis	2.5	0.171	159.2	9	0.171	0.014	Choose
Allocebus_trichotis	Avahi_laniger	1.5	0.103	110.2	10	0.103	0.01	Choose
Allocebus_trichotis	Cheirogaleus_major	3	0.205	76.4	4	0.205	0.024	Choose
Allocebus_trichotis	Cheirogaleus_medius	1	0.068	76.4	4	0.068	0.008	Choose
Allocebus_trichotis	Daubentonia_madagascariensis	2.5	0.171	110.2	8	0.171	0.016	Choose
Allocebus_trichotis	Eulemur_coronatus	1	0.068	110.2	11	0.068	0.007	Choose
Allocebus_trichotis	Eulemur_fulvus	4.5	0.308	110.2	13	0.308	0.029	Choose
Allocebus_trichotis	Eulemur_macaco	5.9	0.404	110.2	13	0.404	0.038	Choose
Allocebus_trichotis	Eulemur_mongoz	1	0.068	110.2	13	0.068	0.007	Choose
Allocebus_trichotis	Eulemur_rubriventer	1.1	0.075	110.2	13	0.075	0.007	Choose
Allocebus_trichotis	Euoticus_elegantulus	0	0	159.2	12	0	0	Choose
Allocebus_trichotis	Galago_alleni	1.5	0.103	159.2	10	0.103	0.008	Choose
Allocebus_trichotis	Galagoides_demidoff	0.5	0.034	159.2	9	0.034	0.003	Choose
Allocebus_trichotis	Galagoides_zanzibaricus	0.5	0.034	159.2	8	0.034	0.003	Choose
Allocebus_trichotis	Galago_moholi	1.2	0.082	159.2	13	0.082	0.007	Choose
Allocebus_trichotis	Galago_senegalensis	0.5	0.034	159.2	13	0.034	0.003	Choose
Allocebus_trichotis	Hapalemur_aureus	0.5	0.034	110.2	12	0.034	0.003	Choose

FIGURE 4.14: A sample output from the summary table in step 3.

With these intuitive displays and interactive screens, a simple, but powerful mechanism is provided to explore the calculated pairwise comparisons and their dependencies. Thus, meaningful and significant comparisons can be found, selected, and analyzed.

4.4.2 Pairing Visualization and Statistics

Pairing Visualization Although it is not mandatory to visualize a pairing, it is very useful to improve general understanding. Especially if the tree contains many species and the number of possible pairings is big, it is hard to imagine how the selected pairs are distributed along the tree.

The user can access the graphical output algorithm from different sites, and each fulfills a particular purpose. Whatever that purpose is, either to visualize a manually specified pairing or the pairing that is automatically determined by the maximal pairing algorithm, this graphical representation greatly helps to see the distribution along the tree. It also becomes immediately clear why other species or pairs of species cannot be selected anymore due to certain constraints.

We implemented an algorithm that creates a graphical output of the results. The following points should be fulfilled to create output that is helpful for the interpretation:

- The tree drawing algorithm must produce optically attractive trees, independent of the number of species. Further details on what properties must be fulfilled to achieve optically attractive trees are omitted here, since that is out of the scope of this thesis.
- Edges should be colored in the following way:
  - Edges that belong to any pairwise comparison in the pairing should be colored, otherwise not.
  - Both species from a pairwise comparison and the connecting path between them should be colored the same.

    - The colors for different pairwise comparisons that belong to the pairing should be distinct from each other and the spectrum of all colors should be scattered as neatly as possible. This is achieved with a simple function that allocates a random color (with the constraint that it must be neither too bright nor too dark) to each selected pairwise comparison and their path on the tree.
    - Edges with a length of zero (reflecting polytomies) should be drawn thinner to distinguish them from non-zero edges.
- The following groups of species should be notably marked:
    - Species that are included in the analysis, but, nevertheless, do not belong to the pairing
    - Species that are excluded from the analysis

We use a recursive algorithm, proceeding from the root of the tree up to the leaves, to draw the phylogenetic tree. This algorithm works as follows:

- For every node, we determine the number of children.
    - If children exist (that is, the node is not a tip), then the space that must be allocated for the particular subtree is calculated. This is dependent on the number of species in this subtree; the bigger this number, the more space must be allocated. We then draw lines (either horizontal and vertical, or diagonal if the node is polytomous) to connect to the descendent subtree. The color of the line is determined by the associated edge object, and this depends on whether the branch is affiliated with a selected pairwise comparison. Then, for each children, we call the recursive procedure again.
    - If no children exist (that is, the node is a tip), one horizontal line is drawn to the right, the name of the species is added and further adjustments (determining if the species is included or excluded, paired or unpaired) are made. The recursion procedure stops in this case, and since the depth of the tree is finite, it is guaranteed that the algorithm stops after a finite number of calls.

Finally, this procedure yields a graphical representation of the phylogenetic tree from the data file, together with a visual overview of all selected pairwise comparisons and their distribution along the tree (see Figure 4.15 for an example).

Pairing Statistics We also provide some basic statistics about properties of the chosen pairing, which include the following:

- The number of pairs that have been selected
- The number of branches that are involved in the pairing
- The maximum number of branches that can be involved in a pairing (this equals the number of branches in the whole tree subtracted by the number of branches that cannot be allocated due to missing data)
- The proportion of the last two elements
- The average number of branches that are involved in each contrast

4.4.3 Target Variable Visualization and Statistics

As highlighted previously, one purpose of the target variable is to determine the distribution of collected data along the tree. This enables the user to easily identify where the missing data points are. We implemented a tree visualization procedure that graphically identified these data points, and it works as follows:

- We draw the regular tree structure together with the information regarding whether a particular species has been studied, which is provided from the target variable.
- Species that have been studied are marked with a red arrow at the right.
- Furthermore, to quickly see how well particular lineages have been studied, every interior node (hereafter called n) that contains more than ten[9] species in the subtree rooted at n has two associated values with it.
    - The first number shows the number of species that have been studied regarding the target variable in the particular subtree that is rooted at n.
    - The second number shows the number of total species in the particular subtree that is rooted at n.

  With this measure, one can quickly identify those lineages in the tree that contain a high proportion of missing data, and an example screenshot is provided in Figure 4.15. Thus, gaps can rapidly be revealed.

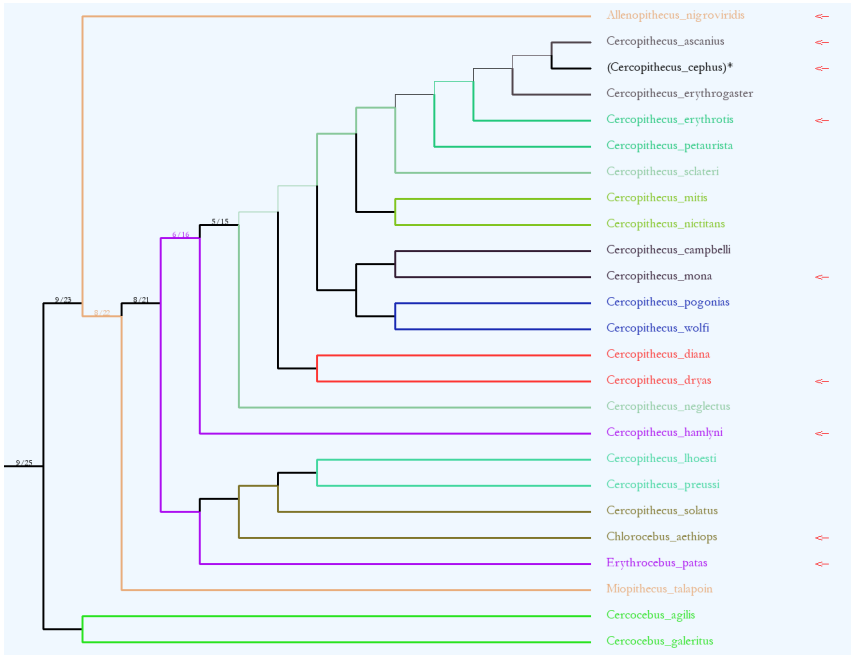


FIGURE 4.15: Example output from a tree with 25 species and 12 selected pairwise comparisons. Every pairwise comparison color is random, and species marked with brackets and * are not selected. All species with a red arrow at the right are species where data are available. See also text for details.

[9] This is the default value as specified in the *TreeDrawer* class. Future versions may have this value as parameter in the web frontend.

4.5 Application Issues

In this section, further issues that belong to the application rather than the methodology itself are presented.

4.5.1 Web Content Accessibility

An important component in the development of the *PhyloTargeting* program was to create a dynamic website that fulfills most points of the latest version of the *W3C Working Draft*, the *Web Content Accessibility Guidelines 2.0*[10]. These guidelines cover a wide range of issues and recommendations for making web content more accessible, and this affects how users complete web-based tasks and find the information or features they want. An accessible interface can help to motivate users to work with the program, and it makes these tasks both easier and more efficient. This should be the main goal for any scientific program, and hence we took this issue into account as well. Following these guidelines creates dynamic web pages with the following properties[11], which are described as follows, as well as implemented measures to accomplish these properties:

1. **"Information and user interface components must be presentable to users in ways they can perceive"**
   - The whole website is divided into hierarchical sections and subsections with descriptive titles.
   - All sections have a logical, clearly arranged structure.
2. **"User interface components and navigation must be operable"**:
   - The use of progress bars for every crucial calculation helps the user to estimate how long the calculation will take. Furthermore, it informs the user that the program is running and that no error occurred.
   - An example file is provided, and this file can be easily chosen in step 1. This is a good starting point for new users to explore the possibilities of the program. The user can also see the format of the NEXUS file that is used.

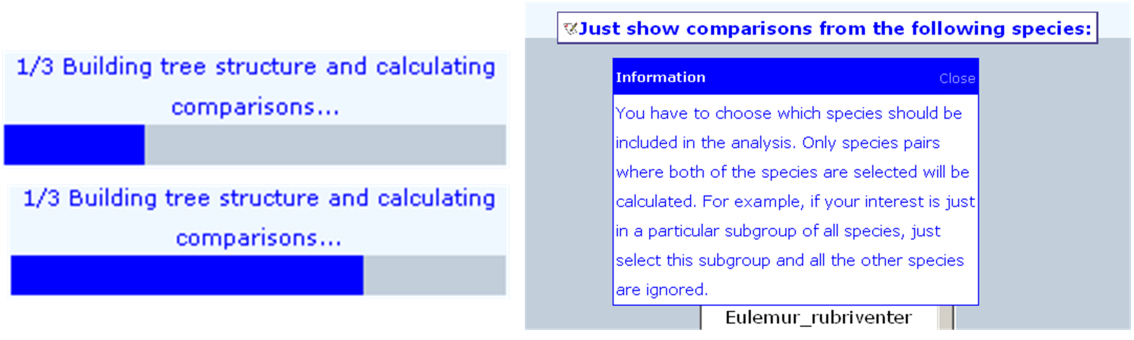


FIGURE 4.16: Screenshots from the progress bar (left two pictures) and an explanation box (right picture).

[10] http://www.w3.org/TR/WCAG20/
[11] http://www.w3.org/TR/2007/WD-WCAG20-20071211/

3. **"Information and the operation of user interface must be understandable"**
   - A two-step 'select and confirm' process is used to reduce accidental selections for critical functions (e.g., in step 2).
   - Explanation boxes are provided for every important element or terminology. This highly improves the understanding of the options. Moreover, the investigator may not need to look in the manual.
   - Users are instructed how to modify selections in critical functions (e.g. with the help toolboxes, see guideline 2)
   - Use page design, graphics, colors, and fonts to clarify complex text and provide summaries to aid understanding (e.g. step 3).
4. **"Content must be robust enough that it can be interpreted reliably by a wide variety of user agents"**

   PhyloTargeting runs on different browser versions, although the design can be slightly different due to different browser implementations of certain technologies (e.g., XHTML and CSS). The functionality is, however, nevertheless guaranteed. We checked the behavior of the site in a range of currently used browser versions: Firefox 2, Internet Explorer 6 and 7, Opera 9 and above, Safari 3. According to Figure 4.17, this selection covers over 98% of the whole browser spectrum.

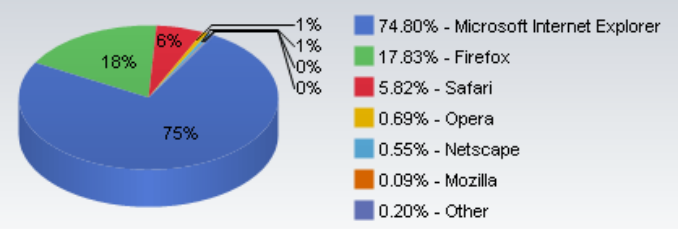


FIGURE 4.17: Browsers Market Share Results, March 2008 [46]

PhyloTargeting uses XHTML and CSS, both are robust and recommended technologies for web programming. Thus, future compatibility should be guaranteed.

4.5.2 Export and Saving

Export Two different export functions are available. They can be accessed from all of the summary tables by clicking the appropriate link in top of the table and are described as follows:

- Every HTML table on the website can be exported to a PDF. This file can then be easily printed out if desired, or used for another purpose. The PDF creation itself is implemented using a free PHP-PDF library, FPDF[12]. Additionally, we implemented some extra features to facilitate reading of the output (e.g., all font sizes are automatically adjusted to fit into the cells and both table header and page numbers appear on every page to guarantee lucidity and clarity).

[12] http://www.fpdf.org/

- In addition to the PDF creation, each HTML table can be exported to a general comma separated text file. This can then easily be imported into spreadsheet programs or statistical packages.

PhyloTargeting (Christian Arnold and Charles Nunn)

Species 1	Species 2	Group_Size	Adjusted diff.	Sum of br. lengths	# of incl. branches	Sum of adj. scores	Stand. raw score
Allocebus_trichotis	Arctocebus_calabarensis	2.5	0.171	159.2	9	0.171	0.014
Allocebus_trichotis	Avahi_laniger	1.5	0.103	110.2	10	0.103	0.01
Allocebus_trichotis	Cheirogaleus_major	3	0.205	76.4	4	0.205	0.024
Allocebus_trichotis	Cheirogaleus_medius	1	0.068	76.4	4	0.068	0.008
Allocebus_trichotis	Daubentonia_madagascariensis	2.5	0.171	110.2	8	0.171	0.016
Allocebus_trichotis	Eulemur_coronatus	1	0.068	110.2	11	0.068	0.007
Allocebus_trichotis	Eulemur_fulvus	4.5	0.308	110.2	13	0.308	0.029
Allocebus_trichotis	Eulemur_macaco	5.9	0.404	110.2	13	0.404	0.038
Allocebus_trichotis	Eulemur_mongoz	1	0.068	110.2	13	0.068	0.007
Allocebus_trichotis	Eulemur_rubriventer	1.1	0.075	110.2	13	0.075	0.007
Allocebus_trichotis	Euoticus_elegantulus	0	0	159.2	12	0	0
Allocebus_trichotis	Galago_alleni	1.5	0.103	159.2	10	0.103	0.008
Allocebus_trichotis	Galagoides_demidoff	0.5	0.034	159.2	9	0.034	0.003
Allocebus_trichotis	Galagoides_zanzibaricus	0.5	0.034	159.2	8	0.034	0.003
Allocebus_trichotis	Galago_moholi	1.2	0.082	159.2	13	0.082	0.007
Allocebus_trichotis	Galago_senegalensis	0.5	0.034	159.2	13	0.034	0.003
Allocebus_trichotis	Hapalemur_aureus	0.5	0.034	110.2	12	0.034	0.003
Allocebus_trichotis	Hapalemur_griseus	0.5	0.034	110.2	12	0.034	0.003
Allocebus_trichotis	Hapalemur_simus	3.5	0.24	110.2	11	0.24	0.023
Allocebus_trichotis	Indri_indri	0.3	0.021	110.2	9	0.021	0.002
Allocebus_trichotis	Lemur_catta	11.6	0.795	110.2	9	0.795	0.076
Allocebus_trichotis	Lepilemur_mustelinus	2.8	0.192	110.2	6	0.192	0.018
Allocebus_trichotis	Loris_tardigradus	1	0.068	159.2	9	0.068	0.005
Allocebus_trichotis	Microcebus_coquereli	2	0.137	54.8	3	0.137	0.019
Allocebus_trichotis	Microcebus_murinus	1.5	0.103	54.8	4	0.103	0.014
Allocebus_trichotis	Microcebus_rufus	1.5	0.103	54.8	4	0.103	0.014
Allocebus_trichotis	Otolemur_crassicaudatus	0.5	0.034	159.2	12	0.034	0.003

Page 1/23

FIGURE 4.18: A sample output from the PDF version of the summary table.

Saving the state of the application For complex and especially time-consuming applications, a very desirable feature is to save the state of the application and continue the analysis at a later date. Such features are implemented to greatly improve the usability.

From an informatics point of view, this is done by serializing the session variable[13] and providing this string to the user. The user is then able to save it as a file to a local hard disk, and at a later date, simply upload it to the website. After that, the exact application state from the previous analysis is restored (see also Figure 4.19). Enhanced security mechanisms need to be incorporated in this procedure, as discussed in the next section.

4.5.3 Security

A wide range of threats face any web application. Especially for PHP, probably the most popular web development language, security is absolutely crucial. In the last two years, there have been numerous security alerts concerning PHP applications [11]. However, the majority of them are not a result of flaws in PHP itself; instead, this is due to developers writing insecure code. PHP is a

[13] which is generated by PHP and automatically stores all the session information

very convenient programming language, often with different ways to achieve a goal. Unfortunately, convenience often weakens security, and that makes PHP a prime candidate for misuse. User friendliness and security safeguards are not completely mutually exclusive, and increasing security often also decreases usability. However, we tried to make a compromise, simply because usability is also an important issue.

From the mass of possible security attacks, only some potential risks remain, and they are briefly discussed next.

1. **Files and Commands**

   User input file has to be validated. This is a crucial step, because the whole program relies on this file. Thus, *PhyloTargeting* needs to be sure that the input file is a compatible NEXUS file. The following approaches to minimize the likelihood that unsupported content may influence the normal behavior of the program have been implemented:

   - The user input file must be a simple text file in the NEXUS format. On account of this, we check if the '#NEXUS' statement appears at the beginning of the file[14]. This simple measure should quickly exclude non-text files.
   - Every mandatory block in the NEXUS file is parsed using regular expressions, and in the case of any errors, the parsing procedure immediately stops and creates a specific error description. This description is then sent to the user, and the temporary uploaded file will be deleted.

2. **Sessions**

   Sessions can be manipulated, and careful consideration is mandatory. The user can always save the state of the application in a downloadable session file; this file can then be reloaded to continue the analysis at a later date. However, this convenience also has risks, because the user can modify the downloaded session file. Measures therefore are needed to detect any modification to the session file, and it should be sufficient to validate the stored session files in a way that no modified version is accepted.

   First, the serialized session content is encrypted using MIME base64[15], which prevents users from reading and easily modifying the file (even though it is still possible to decrypt it). Furthermore, the MD5 hash of the encrypted file is stored in a simple text file (the *hash file*) on the server or local machine. Whenever a user uploads a stored session file, the MD5 hash is determined and the file is only accepted if the hash file contains this particular MD5 hash. As long as this file is protected and nobody except the program itself modifies it, this should allow *PhyloTargeting* to detect[16] modifications of the stored session file. However, the file should not be deleted, since that would prevent any user from continuing a previous analysis.

3. **Further measures**

   - Always validate GET and POST-variables
   - Use *.htaccess* to restrict access to sensitive files

[14] Of course, every text file can be modified in a way that this statement appears first.
[15] http://de.php.net/base64_encode
[16] Strictly speaking, there is a very low probability that a modification will not influence the hash value.

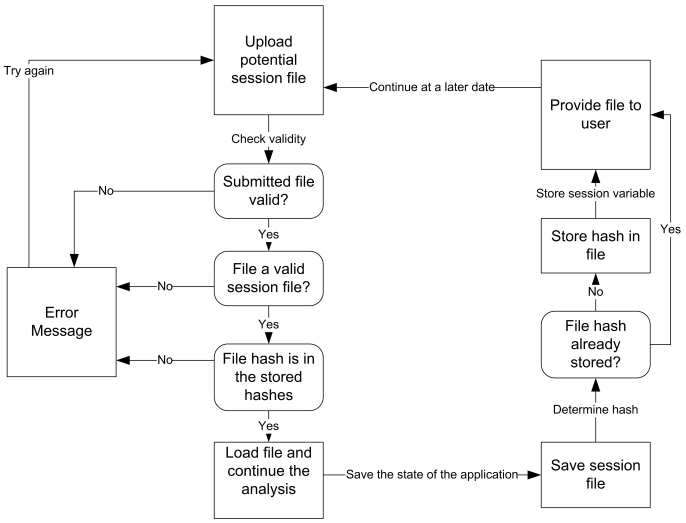


FIGURE 4.19: Flow diagram of the loading and saving procedure.

- We avoid using global variables, because they are a known security risk[16]. This is also particularly important because *register_globals*[17] are deprecated and disappeared in PHP 6.

All of the proposed solutions are implemented in *PhyloTargeting* and the risk of a successful attack is thus greatly minimized.

4.6 Summary and Application Areas

In this chapter, a detailed overview of the new methodology and the developed computer program has been given. Numerous aspects have been discussed, and to summarize, a function overview as well as possible application areas are provided.

4.6.1 Function Overview

The following functions have been implemented:

- Support NEXUS files and automatically detect the relevant information
- Support an arbitrary number of trees and allow users to select and change them whenever they want
- Support an arbitrary number of discrete and continuous traits
- Possibility of selecting one main hypothesis
- Possibility of selecting an arbitrary number of alternative hypotheses, each of which can be scored separately
- Provide different screening mechanisms:
    - Possibility of choosing the species that should be included in the analysis

[17] http://www.php.net/manual/en/security.globals.php

    - Possibility of specifying a target variable
- Generation of all possible pairwise comparisons, together with additional information for each of them
- Provide different scoring mechanisms to discriminate between compelling and less compelling pairs, according to the hypotheses
- Provide an optional branch length standardization method to give all pairwise comparisons a common variance
- Provide a feature that selects a subset of compelling pairwise comparisons (pairing)
    - Provide an algorithm that automatically determines a pairing with maximal score
    - Provide a feature called *contrast selection* to manually specify a pairing
- Provide different visualization mechanisms to present and analyze the results
    - Visual display of all calculated pairwise comparisons in a table (including scores, phylogenetic information, and so forth)
    - Provide an algorithm that graphically displays all selected comparisons in the chosen phylogenetic tree
    - Provide measures that quickly identify clades with a high proportion of missing data as well as the distribution of missing data points in the phylogeny
- Provide further methods to analyze the pairwise comparisons
    - Show only the pairwise comparisons from one selected species
    - Show only selected pairwise comparisons
    - Show only pairwise comparisons that have been automatically determined by the maximal pairing algorithm
    - Every table can be exported to a comma-separated text file or a PDF file
    - Allow arbitrary sorting in every table
- Provide the possibility to save the current application state to a file to allow continuing the analysis at a later date
- Guarantee a basic security level
- Guarantee usability and facilitate the usage of the program through sophisticated help features (help toolboxes)

4.6.2 Application Areas

The program itself neither requires particular data nor exist strong assumptions that limit the applicability. The following aims can be applied to all areas:

- One can quickly disclose species (entities) that conflict the general pattern of a hypothesis.
- One can identify target species that offer the most power to test new hypotheses.
- New hypotheses can be tested for their catholicity.

The approach can be useful to all areas where phylogenetic comparative methods are common. Furthermore, areas that rely on understanding variation among species or other entities are a potential application field. The following overview lists some of them:

- **Anthropology, socioecology and animal behavior**: Comparative methods have been an integral component of these fields since their inception. They have been widely used to study the complex social relationships and diverse ecologies, especially in primates.
- **Conservation biology**: The program might also be useful in this field by picking out species that are most important to study and conserve for future study, including conserving phylogenetic history [17, 63].
- **Bioinformatics**: This field is very widespread, and possible application areas include:
  - Comparative genomics, which studies the relationship between genomes of different species.
  - Analysis of gene, protein or hormone expression or drug development.
  - More generally, optimization problems that rely on comparison of pairs, based on a graph or tree structure, could be solved with the maximal pairing algorithm or a variant of it.

To summarize, the methods developed will be broadly applicable across a number of fields, if appropriate, analyzable data sets are available or can be created.

Chapter 5

Benchmarking and Efficiency

"We all agree on the necessity of compromise. We just can't agree on when it's necessary to compromise."

Larry Wall (1956 - present)

In this chapter, we present and discuss benchmark results as well as implemented measures that improve the overall efficiency of the method.

5.1 Benchmark Methods

Benchmarking an application is an important step to measure the overall performance. This is especially true for web applications, because they are usually a multi-user environment. This may not be critically important if a local version of the application is installed, but, nevertheless, it is also important to consider the performance.

5.1.1 Data Files and Setup

We generated numerous data files which vary from each other in the number of taxa and in the number of traits. The number of taxa varies from 50 to 250, in steps of 50. This should cover the spectrum that is common for comparative databases. For every file, two different numbers of traits are used and analyzed:

- The first contains two traits: One discrete variable (randomly filled with 0 and 1 by Mesquite) and one continuous variable (uniformly random filled with a range of 0 to 50). Furthermore, one main hypothesis (continuous variable) and one alternative hypothesis are used for analysis.
- The latter contains five traits: Four discrete variables and one continuous variable, with the same random fills as in the former case. One main hypothesis (continuous variable) and four alternative hypotheses are used for analysis.

Both files are based on a perfectly balanced tree without polytomies, and the data contains no missing information. Thus, these measurements can be seen as best cases, since the average performance decreases when the balancing of the tree decreases (see Chapter 4). However, for average balanced trees, the results should be similar.

Benchmarks for execution times reported here are an arithmetic mean of three independent measurements for each set of taxa[1], and they do not cover the time to deserialize the session file or other issues. We focused on two measures of execution time: The time needed to generate and initialize the data structures and the execution time for the maximal pairing algorithm. We also report the size of the session file that has to be stored permanently and the maximal memory usage[2]. To make it comparable, we also arbitrarily excluded a set of species for every data file to limit the number of pairwise comparisons in order to compare execution times among different data sets. This controls if execution time correlates with the number of taxa.

5.1.2 Results

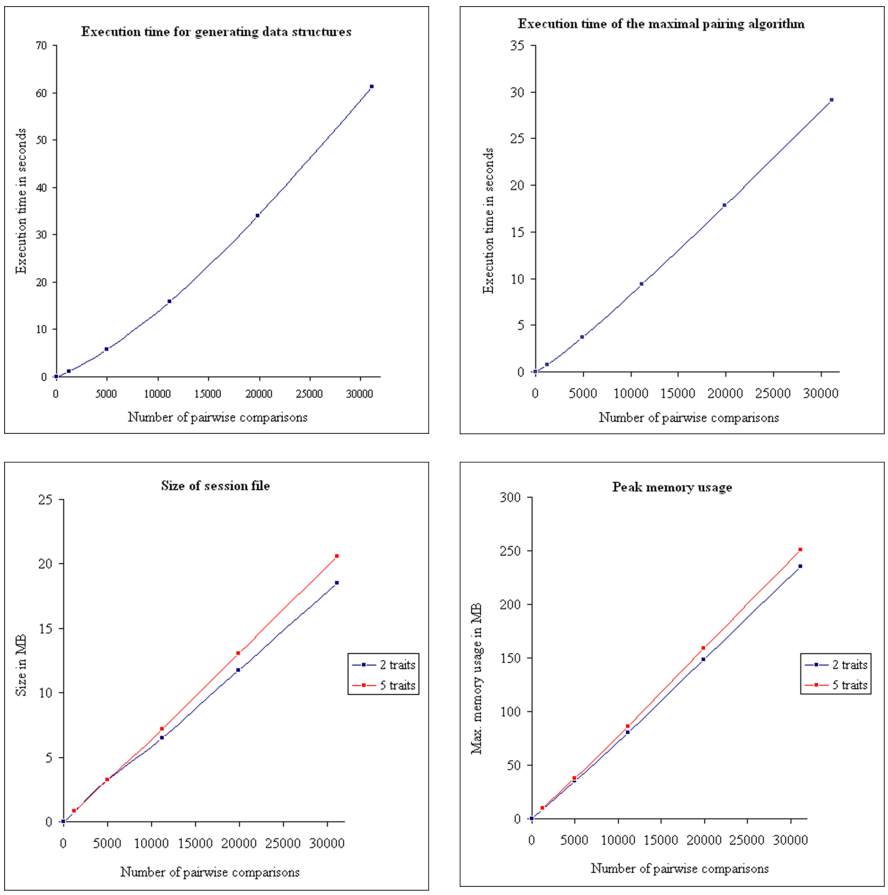


FIGURE 5.1: Different diagrams that summarize the benchmark results. All diagrams are in dependence of the number of pairwise comparisons that must be generated in the dataset. For complete datasets, as in this simulation, 50 species equals 1125 comparisons, 100 species equals 4950 comparisons, 150 species equals 19900 comparisons, and 250 species equals 31125 comparisons. The discrimination between two and five traits has been omitted for two diagrams due to the fact that no difference in execution time was detectable. See also text for details.

[1] The program ran on a Pentium M 1.6 GHz with 512 MB RAM.

[2] This is measured using the *memory_get_peak_usage* (http://de2.php.net/memory_get_peak_usage) function. This function returns the peak of memory that has been allocated to a PHP script and includes also the amount of memory needed for the Apache webserver. Thus, this is only an indirect measure, because it is difficult to predict which amount the PHP script truly needs.

As shown in Figure 5.1, we created different diagrams to visualize the benchmark results, and the diagrams show the following results:

- The number of traits does not influence the execution times (see above), and the differences in the size of the session file are small compared to the size of the session file itself. The same holds for peak memory usage. Thus, the number of traits is not that important for the general performance of the application.
- The size of the session file and the allocated memory can become very high when the number of pairwise comparisons is large. However, we do not expect such large datasets, because a lot of missing information is expected in most cases.
- The execution time to generate and initialize the data structures does not grow linearly with the number of pairwise comparisons. However, it should be fast enough for usage.
- For perfectly balanced trees, the maximal pairing algorithm shows an increase in execution time with increasing number of pairwise comparisons, as predicted in Chapter 4.

We also measured the execution time for the maximal pairing algorithm in the case of polytomous nodes. Specifically, we measured the execution times for polytomous trees with a polytomous node of degree 3 to 12 as root, whereas all other nodes are dichotomous. As we already concluded in Chapter 4, the execution time of the maximal pairing algorithm is exponential to the degree of the node. This should yield to a doubled execution time whenever the degree of polytomy is increased by one. Indeed, as Figure 5.2 shows, we can observe such an exponential increase.

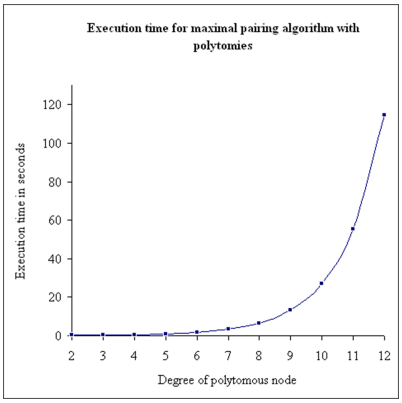


FIGURE 5.2: Execution time of the maximal pairing algorithm for polytomous trees. See text for details.

5.2 Measures for Improving the Overall Efficiency

There are many ways to improve the efficiency of a program. Sometimes, execution time and memory usage are not an issue, even with large data sets. However, in most cases, they are. Web applications face the additional problem that an arbitrary number of users may use them at the same time. Therefore, appropriate data structures, efficient memory management, and fast algorithms should be used. Nevertheless, all measures should also be in proportion to the effort. A compromise between memory allocation and execution time had to be found, and neither of the

two can be neglected. This section briefly describes this challenge.

5.2.1 Data Structures and Execution Time

The most decisive question in the implementation of the program was to choose among different data structures. Momentous improvements have been made by this decision as well as with the help of numerous measures that improve the execution time. Some basic ideas are illustrated as follows:

- The underlying data structure that is used to represent the complex interactions is a tree structure. In contrast to earlier implementations of the program, significant speed improvements have been achieved using this kind of data structure, as compared to regular non-treelike structures.
- By saving as little information as possible, a dramatic reduction of the execution time can be achieved. As mentioned in Chapter 4, the deletion of uninformative pairwise comparisons greatly reduces the runtime of different algorithms (maximal pairing algorithm, sorting, traversing the arrays and searching).
- Another crucial measure was the usage of associative arrays, instead of non-associative ones, which enables the program to achieve $\mathcal{O}(1)$ lookup times with efficient implementations. Lookups are very frequently used in the program, especially in more complex algorithms like the maximal pairing algorithm. It is therefore essential to achieve minimum lookup times. For large arrays, an extraordinary speedup is achieved by implementing this simple measure.
- The maximal pairing algorithm has one further preprocessing step, in addition to the ones described in Chapter 4, that greatly reduce the execution time if a lot of missing information is present. For most datasets, we indeed expect a high proportion of missing data. Species that have missing data in one of the traits of interest are not valuable, and can therefore be excluded. This is also true for interior nodes if they have exclusive nodes as descendants that are also excluded. Thus, we can apply a recursive algorithm that checks all interior nodes, and in the case of exclusion, the score can be immediately set to 0. This improves the execution time of the maximal pairing algorithm, especially in the case of polytomies (because the exclusion basically means a reduction of the degree of the node).

5.2.2 Memory Management

Memory management is of great importance for the developed application because the needed structures can become very large and complex. The storage of all pairwise comparisons needs a huge amount of memory, and this increases quadratically with the number of species. Hence, we must be extremely careful of how the data are stored and represented. Furthermore, all of the data is saved in the session file, and thus this file can become very large. This has dramatic effects: The page load increases, because we have to unserialize this session file on every page, more memory on the hard disk is needed and finally, a larger session file leads to more network traffic. However, the application does necessarily allocate a lot of memory, due to the fact that a lot of information must be stored persistently. This cannot be avoided, and great effort has been made to limit this

constraint as much as possible. The following mechanisms have been developed to allocate less memory:

- Objects should contain as few attributes as possible. This is particularly true for the *PhyloComparison* object, because numerous instances of this object type typically must be stored.
- Memory that must be allocated only temporarily (e.g., for the preprocessing step in the maximal pairing algorithm) is freed afterwards.
- Data that are not needed for a specific set of settings are neglected. Specifically, the following data are deleted:
  - All pairwise comparison objects that are not informative to the user-specified settings. This measure had the biggest effect on memory saving if a high degree of missing data is present or if constraints (e.g., target variable) exclude a lot of pairs.
  - Trait differences that do not represent the hypotheses of interest.

5.3 Directions of Further Improvement

- The determination of the last common ancestor node can be further improved by using more sophisticated data structures such as suffix trees [see Gusfield [28] for details] to achieve $\mathcal{O}(1)$ algorithms with less preprocessing time.
- The maximal pairing algorithm for polytomous nodes can be improved as follows:
  - The complexity can be reduced to a polynomial degree using the Gabow [21] algorithm as briefly described in Chapter 4.
  - Execution time can be improved if we saved all leftover subtrees permanently for all pairwise comparison objects that have a polytomous node as last common ancestor. This is due to the fact that the subtree identification procedure is called multiple times in case of polytomies, which is redundant. However, for memory reasons, we did not implement this measure.
- Memory usage and the serialization or deserialization process are bottlenecks for large data files. The application can easily allocate a lot of memory for large data sets. The main reason for this high memory usage is the fact that we need to store all edge IDs that a particular pairwise comparison allocates for the *contrast selection* procedure to make the feature usable. Future versions may change this implementation, which would dramatically decrease the memory usage. Thus, although many techniques have been implemented that reduce this amount as much as possible, space for further improvement is present.

Chapter 6

Real-World Application

"The incorrectness and weaknesses of a theory cause other minds to formulate the problems more exactly and in this way scientific progress is made."

Robert Bárány (1876 - 1936)

In this section, we apply a real-life dataset to the *PhyloTargeting* approach to test its effectiveness and applicability.

6.1 Introduction to Sleep

Sleep is an evolutionary puzzle and the functions of sleep are not immediately apparent. It can be defined as a state of natural rest that is observed in most mammals, birds, fish, and invertebrates. Sleep is an extraordinary complex phenomenon, consisting of environmental, psychological, physiological and behavioral components that influence sleep durations, the number of sleep bouts per day, and the intensity with which animals sleep. The phylogenetic context of sleep has been studied widely [9, 39, 1, 75], and these phylogenetic analyses raise many interesting questions concerning the function and evolution of sleep: What are the relationships between ecology, life history, behavior, physiology, and patterns of sleep? What factors account for whether animals sleep in one bout per day (monophasic) or multiple bouts (polyphasic)? What are the benefits of sleep, and can these be assessed through broad-scale comparisons?

To answer all these questions, data are needed to test hypotheses related to the function and evolution of sleep. But from which species should the data come? As already mentioned in Chapter 1, huge gaps in the distribution of studied species exist. To overcome these biases, which would provide a more general understanding of sleep's function (or functions), we need to study more species. The identification of species that are compelling to test new hypotheses could result in new insights into sleep, and as noted in [54], "it is critically important to fill in some of the gaps in our knowledge of primate sleep, and to do so in a way that provides the strongest tests of comparative hypotheses.". In what follows, we apply the *PhyloTargeting* framework to this fundamentally important question.

6.2 Dataset and Hypotheses

We collected data on body mass, brain mass, and activity period (a discrete variable describing whether the species is nocturnal or diurnal) for 72 primate species. We also included a discrete trait indicating whether a species has been studied for sleep, based on the 'Phylogeny of sleep' [50] database. Brain and body mass are the weighted average of male and female individuals, both of which may consist of more than one individual. The number of measurements per species ranged from 1 to 54. The phylogenetic tree is based on a recently published supertree [4] and consists of 233 taxa. Thus, we have missing data for 161 species.

The reasoning for using these variables is as follows. Sleep is often thought to be beneficial for the brain (memory consolidation). Comparative studies, however, have revealed only mixed support for this hypothesis. It might be the case that the species studied have not offered the strongest tests for this hypothesis. Thus, it would be worthwhile to target species that differ in brain size. For this, we need to control for body mass, because brain and body mass are strongly correlated. To calculate a measure of relative brain size, we regressed brain mass on body mass and calculated the residuals; this was done using phylogenetically independent contrasts [18] in the computer program Mesquite [45] and the PDAP package [51] within Mesquite. The exact procedure is described in Appendix C, because the use of such phylogenetic comparative methods is beyond the scope of this thesis.

A second hypothesis is whether body mass and sleep are linked. Phylogeny based studies have failed to support such a link, whereas non-phylogenetic studies revealed support. Here, we used adult body mass from animals in the wild, whereas brain mass data controlled for body mass of animals in the laboratory that provided brain data (and included some juveniles).

A third hypothesis for which evidence was recently uncovered is that nocturnal species sleep longer than diurnal species. Unfortunately, not enough species have been examined to determine what might drive this, or if the pattern is general.

6.3 Application to the *PhyloTargeting* Framework

According to the 'Phylogeny of sleep' database, the following 20 species have been already studied regarding sleep data:

Scientific name	Common name
Aotus trivirgatus	Northern night monkey
Callithrix jacchus	White-tufted-ear marmoset
Chlorocebus aethiops	Vervet monkey
Erythrocebus patas	Patas monkey
Eulemur macaco	Black lemur
Eulemur mongoz	Mongoose lemur
Homo sapiens	Human
Macaca arctoides	Stump-tailed macaque
Macaca mulatta	Rhesus monkey
Macaca nemestrina	Pigtail macaque
Macaca radiata	Bonnet macaque
Macaca sylvanus	Barbary macaque
Microcebus murinus	Gray mouse lemur
Pan troglodytes	Chimpanzee
Papio hamadryas	Hamadryas baboon
Perodicticus potto	Potto
Phaner furcifer	Fork-marked lemur
Saguinus oedipus	Cotton-top tamarin
Saimiri sciureus	South American squirrel monkey
Theropithecus gelada	Gelada baboon

TABLE 6.1: Species with data available on sleep durations and their common names.

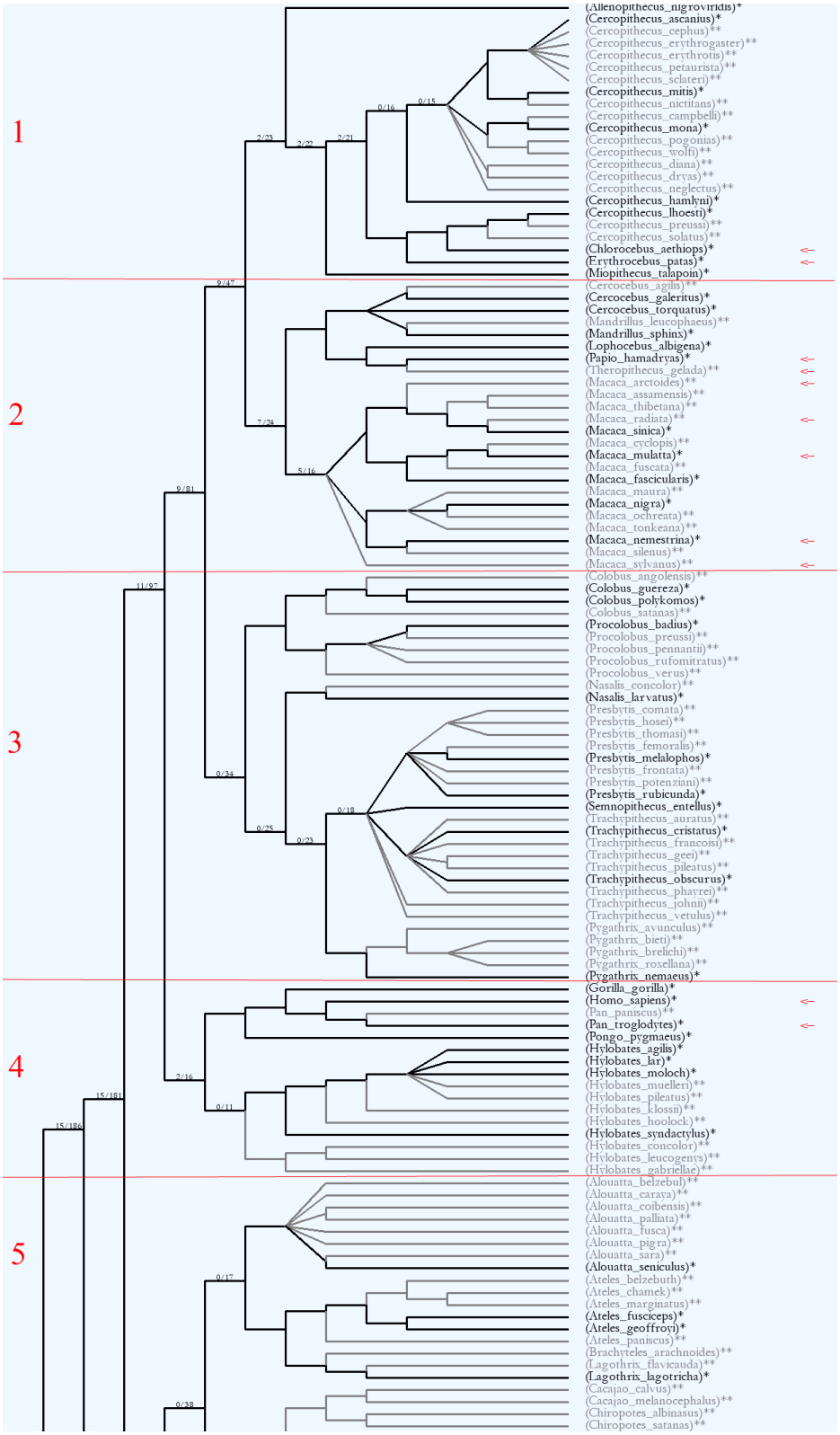
Although not apparent initially, these studied species are gapped in a way that some clades are completely missing, whereas others are well-studied.

We propose the following procedure to identify key species that fill in some of these gaps:

1. Identify clades that either contain a high proportion of missing data or that are interesting in some other way (e.g., the great apes, which includes humans)
2. Apply different targeting analyses to the program, where a targeting analysis refers to a particular set of main and alternative hypotheses, as well as specification of target variables and branch length controls. These different targeting analyses are likely to focus on a primary main hypothesis and various combinations of alternative hypotheses, but could also include different primary hypotheses, depending on the particular goals of the study.
3. For each targeting analysis, calculate the maximal pairing. The species from these maximal pairings can be classified as potential candidates, since they represent the set of species that provide the strongest tests of hypotheses with the specified settings.
4. Consider the information content by calculating a standardized overall score for each species pair in each targeting analysis. Also collate data on the occurrence of particular species pairs across the targeting analyses.

5. With these standardized scores across targeting analyses, calculate the *summed pairing score* (see later in this section for more details). The species or species pairs that have the best summed pairing score can then be classified as key species, since they represent species that provide the strongest tests for a variety of evolutionary hypotheses.

The first step is to identify clades that contain a high proportion of missing data. With the measures provided by the *PhyloTargeting* framework (e.g., target variable), we are able to quickly identify clades that are worthwhile for further investigation. Figure 6.1 shows these interesting clades (taken from the *PhyloTargeting* program, clade numbers has been added for clarity), and all of them are phylogenetically separate and thus non-nested. Studied species are marked with a red arrow at the right, species where data are available are showed in black, and species where no data are available are showed in gray. Furthermore, the phylogeny has been partitioned into nine major clades to facilitate analysis and discussion.


1
2
3
4
5
(Allenopithecus_nigroviridis)*
(Cercopithecus_ascanius)*
(Cercopithecus_cephus)**
(Cercopithecus_erythrogaster)**
(Cercopithecus_erythrotis)**
(Cercopithecus_petaurista)**
(Cercopithecus_sclateri)**
(Cercopithecus_mitis)*
(Cercopithecus_nictitans)**
(Cercopithecus_campbelli)**
(Cercopithecus_mona)*
(Cercopithecus_pogonias)**
(Cercopithecus_wolfi)**
(Cercopithecus_diana)**
(Cercopithecus_dryas)**
(Cercopithecus_neglectus)**
(Cercopithecus_hamlyni)*
(Cercopithecus_lhoesti)*
(Cercopithecus_preussi)**
(Cercopithecus_solatus)**
(Chlorocebus_aethiops)*
(Erythrocebus_patas)*
(Miopithecus_talapoin)*
(Cercocebus_agilis)**
(Cercocebus_galeritus)*
(Cercocebus_torquatus)*
(Mandrillus_leucophaeus)**
(Mandrillus_sphinx)*
(Lophocebus_albigena)*
(Papio_hamadryas)*
(Theropithecus_gelada)**
(Macaca_arctoides)**
(Macaca_assamensis)**
(Macaca_thibetana)**
(Macaca_radiata)**
(Macaca_sinica)*
(Macaca_cyclopis)**
(Macaca_mulatta)*
(Macaca_fuscata)**
(Macaca_fascicularis)*
(Macaca_maura)**
(Macaca_nigra)*
(Macaca_ochreata)**
(Macaca_tonkeana)**
(Macaca_nemestrina)*
(Macaca_silenus)**
(Macaca_sylvanus)**
(Colobus_angolensis)**
(Colobus_guereza)*
(Colobus_polykomos)*
(Colobus_satanas)**
(Procolobus_badius)*
(Procolobus_preussi)**
(Procolobus_pennantii)**
(Procolobus_rufomitratus)**
(Procolobus_verus)**
(Nasalis_concolor)**
(Nasalis_larvatus)*
(Presbytis_comata)**
(Presbytis_hosei)**
(Presbytis_thomasi)**
(Presbytis_femoralis)**
(Presbytis_melalophos)*
(Presbytis_frontata)**
(Presbytis_potenziani)**
(Presbytis_rubicunda)*
(Semnopithecus_entellus)*
(Trachypithecus_auratus)**
(Trachypithecus_cristatus)*
(Trachypithecus_francoisi)**
(Trachypithecus_geei)**
(Trachypithecus_pileatus)**
(Trachypithecus_obscurus)*
(Trachypithecus_phayrei)**
(Trachypithecus_johnii)**
(Trachypithecus_vetulus)**
(Pygathrix_avunculus)**
(Pygathrix_bieti)**
(Pygathrix_brelichi)**
(Pygathrix_roxellana)**
(Pygathrix_nemaeus)*
(Gorilla_gorilla)*
(Homo_sapiens)*
(Pan_paniscus)**
(Pan_troglodytes)*
(Pongo_pygmaeus)*
(Hylobates_agilis)*
(Hylobates_lar)*
(Hylobates_moloch)*
(Hylobates_muelleri)**
(Hylobates_pileatus)**
(Hylobates_klossii)**
(Hylobates_hoolock)**
(Hylobates_syndactylus)*
(Hylobates_concolor)**
(Hylobates_leucogenys)**
(Hylobates_gabriellae)**
(Alouatta_belzebul)**
(Alouatta_caraya)**
(Alouatta_coibensis)**
(Alouatta_palliata)**
(Alouatta_fusca)**
(Alouatta_pigra)**
(Alouatta_sara)**
(Alouatta_seniculus)*
(Ateles_belzebuth)**
(Ateles_chamek)**
(Ateles_marginatus)**
(Ateles_fusciceps)*
(Ateles_geoffroyi)*
(Ateles_paniscus)**
(Brachyteles_arachnoides)**
(Lagothrix_flavicauda)**
(Lagothrix_lagotricha)*
(Cacajao_calvus)**
(Cacajao_melanocephalus)**
(Chiropotes_albinasus)**
(Chiropotes_satanas)**

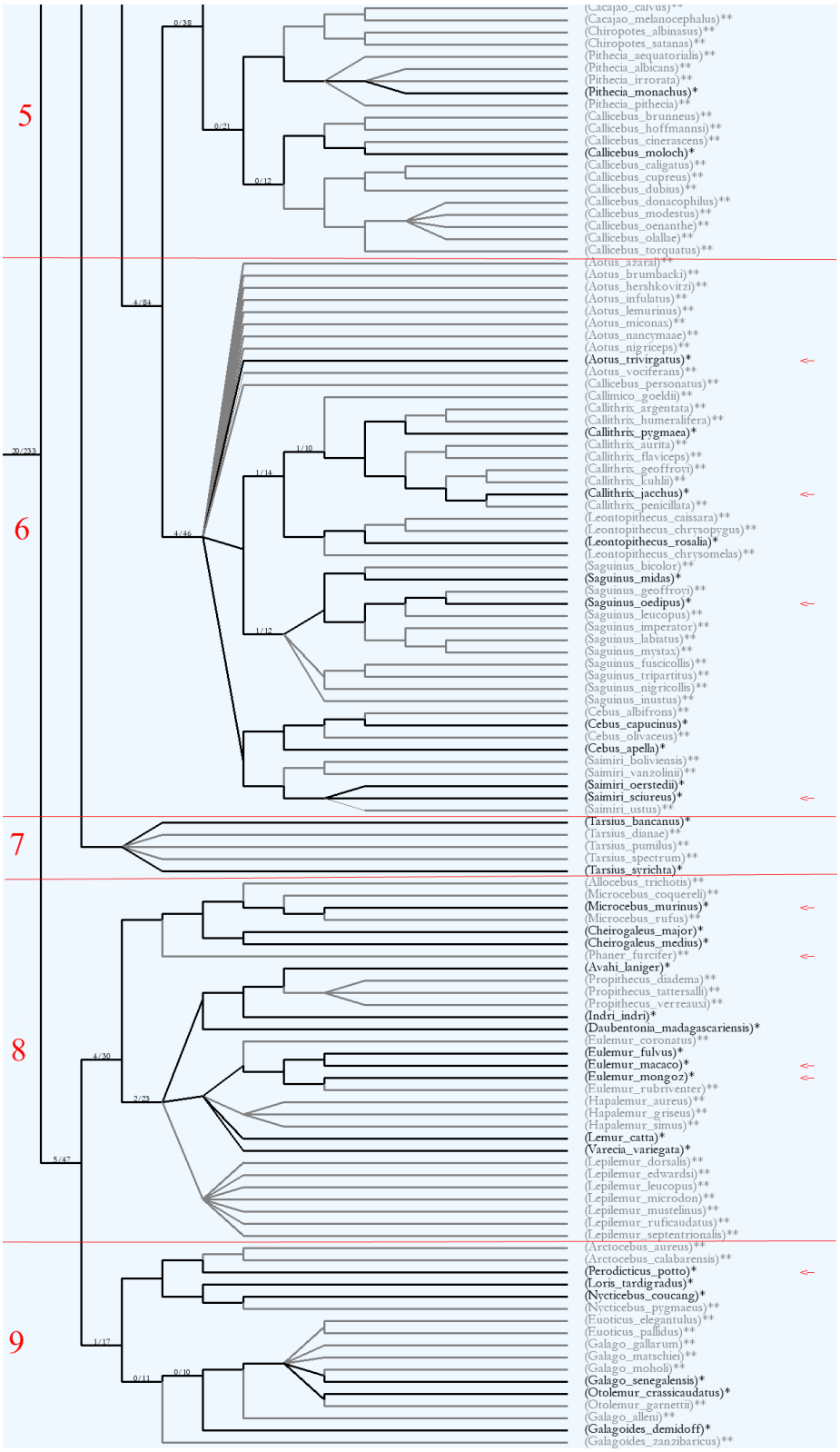


FIGURE 6.1: Graphical tree representation of the phylogeny used in the dataset. See text for details.

In summary, we can extract the following information from this graphical tree representation:

Clade	Proportion studies species / total species in this clade	Studied species in this clade
1	2 / 23 (8.7 %)	*Chlorocebus aethiops*, *Erythrocebus patas*
2	7 / 24 (29.1 %)	*Macaca arctoides*, *Macaca mulatta*, *Macaca nemestrina*, *Macaca radiata*, *Macaca sylvanus*, *Theropithecus gelada*, *Papio hamadryas*
3	0 / 34 (0 %)	/
4	2 / 16 (12.5 %)	*Homo sapiens*, *Pan troglodytes*
5	0 / 38 (0 %)	/
6	4 / 46 (8.7 %)	*Aotus trivirgatus*, *Callithrix jacchus*, *Saguinus oedipus*, *Saimiri sciureus*
7	0 / 5 (0 %)	/
8	4 / 30 (13.3 %)	*Eulemur macaco*, *Eulemur mongoz*, *Microcebus murinus*, *Phaner furcifer*
9	1 / 17 (5.9 %)	*Perodicticus potto*

Table 6.2: Non-nested clades that will be further examined. See text for details.

Now, it is immediately apparent that the distribution of studied species is not homogeneous. Some clades are relatively well studied (e.g., clade 2, especially macaques), whereas other clades are completely missing (e.g., clades 3 and 5). This suggests that collection of data on sleep is heavily biased towards particular species. Collecting data on sleep requires that animals be brought into the lab and acclimated to laboratory conditions, as well as expenses related to acquiring data using EEG. Given the costs of collecting such data, we need a way to systematically identify the species that offer the strongest tests of adaptive hypotheses.

The second step is to apply different targeting analyses and to calculate the maximal pairing. The following analyses are applied to the *PhyloTargeting* framework:

Analysis	Main hypothesis	Alternative hypotheses
1	Relative brain mass	/
2	Relative brain mass	Body mass (no change)
3	Relative brain mass	Activity period (no change)
4	Relative brain mass	Body mass (no change), activity period (no change)

Table 6.3: All four targeting analyses that have been examined using the *PhyloTargeting* program.

Contrast standardization is turned off in all scenarios, since we are mainly interested in the total change, independent of the evolutionary time between these changes. Furthermore, in all scenarios, the target variable is set to a discrete trait indicating whether the species has already been studied. As target variable option, we choose the constraint to consider only pairwise comparisons where at least one species in the pair has missing data according to the target variable.

The third step is worth explaining in more detail. As mentioned in Chapter 4, pairing scores cannot be compared directly across targeting analyses. To account for this, we will introduce a standardized version of this pairing score, which helps to consider the information content from all analyses equally. This standardization works exactly as the scoring scheme from the main hypothesis (see section 4.3.5), thus transforming the raw scores from the maximal pairing for each species pair to the interval $[0, 1]$ by simply dividing by the highest raw score in the maximal pairing. By fixing the minimum to 0, we also guarantee that all species pair in the set obtain a positive score after the standardization.

After summing up all these standardized pairing scores for each species pair in each targeting analyses, we obtain the *summed pairing score*. This score can be used to decide which species are referred to as key species. The range of this score is between 0 and the number of maximal pairings that have been considered, in our case thus between 0 and 4. This concept has two main advantages: It accounts how often a particular species pair occurred, as well as how meaningful, compared to the best-scoring one, these different pairwise comparisons are. For example, if a species pair occurred in all four maximal pairings due to topology reasons (sister species, phylogenetically 'isolated', and so on), but the specific scores are small compared to the maximal score in this set, then we do not want to consider this species pair as compelling. Another example would be if a species pair occurred only a few times, but always with a very high score; this would also lead to a good *summed pairing score*, which is desirable.

6.4 Results

In what follows, we provide statistics about all five maximal pairings from the targeting analyses, since these maximal pairings form the basis for recommending which species are most worthwhile to study in the future.

Analysis	**Pairing score**	**Number of pairs**	**Number of involved branches / Maximum number of branches**	**Number of average branches involved in each contrast**
1	5.17	29	145 / 206 (0.704)	5
2	38.2	35	159 / 206 (0.772)	4.54
3	38.35	34	159 / 206 (0.772)	4.68
4	70.71	34	159 / 206 (0.772)	4.68

TABLE 6.4: Properties of all four maximal pairings, taken from the *PhyloTargeting* program.

In total, 2451 pairwise comparisons are informative according to the data. Due to the fact that we have a high percentage of missing data, this number is small compared to the number of species in the dataset. From these 2451 pairwise comparisons, the maximal pairing algorithm identified those 29-34 (see 6.4) phylogenetically separate pairs that have the highest pairing score.

It is interesting that analyses 2-4 yield to very similar maximal pairings, whereas analysis 1 shows a significantly reduced number of pairs. One reason for that could be that in analysis 1, only one variable was considered (relative brain size). Thus, higher differences automatically lead to

a better score. Higher differences, however, appear mostly in more distant comparisons, which explains both the decreased number of pairs and the smaller proportion of involved branches.

We now report results from the analysis, in which we calculated the *summed pairing scores* for all species pairs that occurred in any of the four maximal pairings.

Species 1	**Species 2**	**Number of occurrences**	**Summed pairing score**
Alouatta seniculus (5)	*Lagothrix lagotricha* (5)	4	3.28
Gorilla gorilla (4)	*Homo sapiens** (4)	3	2.95
Tarsius bancanus (7)	*Tarsius syrichta* (7)	4	2.65
Cercopithecus ascanius (1)	*Cercopithecus mitis* (1)	4	2.58
Avahi laniger (8)	*Daubentonia madagascariensis* (8)	3	2.57
Eulemur fulvus (8)	*Eulemur macaco** (8)	4	2.50
Saimiri oerstedii (6)	*Saimiri sciureus** (6)	4	2.45
Saguinus midas (6)	*Saguinus oedipus** (6)	4	2.41
Galagoides demidoff (9)	*Perodicticus potto** (9)	3	2.36
Presbytis melalophos (3)	*Presbytis rubicunda* (3)	4	2.35
Trachypithecus cristatus (3)	*Trachypithecus obscurus* (3)	4	2.35
Callicebus moloch (5)	*Pithecia monachus* (5)	4	2.35
Loris tardigradus (9)	*Nycticebus coucang* (9)	4	2.34
Nasalis larvatus (3)	*Procolobus badius* (3)	3	2.29
Galago senegalensis (9)	*Otolemur crassicaudatus* (9)	3	2.28
Cercopithecus hamlyni (1)	*Cercopithecus mona* (1)	3	2.23
Macaca fascicularis (2)	*Macaca mulatta** (2)	4	2.22
*Macaca nemestrina** (2)	*Macaca nigra* (2)	3	2.19
Pygathrix nemaeus (3)	*Semnopithecus entellus* (3)	4	2.15
Cercopithecus lhoesti (1)	*Chlorocebus aethiops** (1)	3	2.15
Lophocebus albigena (2)	*Papio hamadryas** (2)	3	2.15
Colobus guereza (3)	*Colobus polykomos* (3)	3	2.15
Ateles fusciceps (5)	*Ateles geoffroyi* (5)	4	2.15
*Erythrocebus patas** (1)	*Miopithecus talapoin* (1)	3	2.14
*Callithrix jacchus** (6)	*Callithrix pygmaea* (6)	3	2.11

TABLE 6.5: Overview of the species pairs from all four maximal pairings. Information on the species pair itself, including the clades the species belong to (in brackets after the name of the species), the number of occurrences in all four maximal pairings, and the *summed pairing score* (rounded to two digits after the decimal point) is provided. A '*' symbol after the species name, as in the *PhyloTargeting* program, indicates that the species has been studied concerning sleep durations. The table is sorted descending after the *summed pairing score* and only the 25 species pairs are listed that occurred more than twice, due to space restrictions.

The 25 species pairs show some interesting patterns: They are all phylogenetically separate and the pairs are all within a particular clade; no species pair belongs to more than one clade.

To pick out the species that should receive highest priority, we summarize the results from the previous table:

Clade	Number of high-scoring species pairs	Species pairs
1	4	*Cercopithecus ascanius- Cercopithecus mitis* (4) *Cercopithecus hamlyni- Cercopithecus mona* (16) *Cercopithecus lhoesti - Chlorocebus aethiops** (20) *Erythrocebus patas** - *Miopithecus talapoin* (24)
2	3	*Macaca fascicularis - Macaca mulatta** (17) *Macaca nemestrina** - *Macaca nigra* (18) *Lophocebus albigena - Papio hamadryas** (21)
3	5	*Presbytis melalophos - Presbytis rubicunda* (10) *Trachypithecus cristatus - Trachypithecus obscurus* (11) *Nasalis larvatus - Procolobus badius* (14) *Pygathrix nemaeus - Semnopithecus entellus* (19) *Colobus guereza - Colobus polykomos* (22)
4	1	*Gorilla gorilla - Homo sapiens** (2)
5	3	*Alouatta seniculus - Lagothrix lagotricha* (1) *Callicebus moloch - Pithecia monachus* (12) *Ateles fusciceps - Ateles geoffroyi* (23)
6	3	*Saimiri oerstedii - Saimiri sciureus** (7) *Saguinus midas - Saguinus oedipus** (8) *Callithrix jacchus** - *Callithrix pygmaea* (25)
7	1	*Tarsius bancanus - Tarsius syrichta* (3)
8	2	*Avahi laniger - Daubentonia madagascariensis* (5) *Eulemur fulvus - Eulemur macaco** (6)
9	3	*Galagoides demidoff - Perodicticus potto** (9) *Loris tardigradus - Nycticebus coucang* (13) *Galago senegalensis - Otolemur crassicaudatus* (15)

TABLE 6.6: Summary of the identified high-scoring species pairs, in relation to their clade. Species pairs in column three are sorted descending by their *summed pairing score*, and information is provided on the rank of the pair from Table 6.5. See also text for details.

Based on Table 6.6, we picked out species that we denote as key species for future study. Some clades are particularly important, as follows:

- The distribution of studied species shows remarkable gaps in clades 3 and 5.
- Moreover, clade 4 (the greater and lesser apes) should receive more attention, since we collected only information for one species of non-human ape (chimpanzee) so far and thus, all of our knowledge about sleep in apes is based on only chimpanzees and humans.

Following these guidelines, we want to point out five species that should, in our opinion, receive highest priority for further data collection, since they both significantly fill in some of the gaps and offer strong tests for evolutionary hypotheses. These five species are the following:

Scientific name	Common name
Alouatta seniculus	Red howler monkey
Gorilla gorilla	Western gorilla
Lagothrix lagotricha	Humboldt's woolly monkey
Presbytis melalophos	Mitred leaf monkey
Presbytis rubicunda	Red leaf monkey

TABLE 6.7: Identified key species and their common names. The table was sorted by column 1.

Another factor that is worth to be considered is that we expect comparisons where one of the species has already been studied to be more important, because one would only need data for one species in order to directly compare the two species that form the pair. The following five species therefore also offer strong power to test hypotheses for the evolution of sleep:

Scientific name	Common name
Cercopithecus lhoesti	L'hoest's monkey
Eulemur fulvus	Brown lemur
Galagoides demidoff	Prince Demidoff's bushbaby
Saguinus midas	Midas tamarin
Saimiri oerstedii	Central American squirrel monkey

TABLE 6.8: Species that also offer strong power, see text for details. The table was sorted by column 1.

Summary

Based on different setups and analysis, we identified ten species that we denote as key species for testing hypotheses that link cognitive demands to sleep durations [39]. Five of them are based on general patterns, and five are based on other species that have been studied. Moreover, they significantly reduce the gap bias in studying sleep in primates.

Chapter 7

Discussion

"The important thing is not to stop questioning."

Albert Einstein (1879 - 1955)

Although the method of pairwise comparisons as a basis for identifying target species has proved to be very useful, unsolved problems still exist. In what follows, we consider both outstanding problems and possible extensions, including extensions that would overcome some of the acknowledged problems.

7.1 Outstanding Problems

7.1.1 Sampling Error and Within-Species Variation

Comparative methods usually assume that species values are the true means for those species. In reality, species values are often only estimates due to samples of modest size, making them subject to random error. Independent contrasts methods are sensitive to measurement and sampling error [20, 31, 36], and data quality is crucial for these kinds of analyses. This sampling error matters less if distant relatives are compared, because it is small relative to the evolved difference since they split. However, when close relatives are compared (which is usually true for the method of pairwise comparisons), the sampling error can overwhelm the small differences between those species. Furthermore, the method of phylogenetically independent contrasts assumes that the phenotypic means of the characters are observed in each species, rather than the means of (often small) finite samples [20]. Within-species variation should be negligible; however, this is a strong assumption and can lead to serious biases. For example, Ricklefs and Starck [72] presented an example where the contrasts with the biggest differences are those that come from closely related species, and they concluded that within-species variation caused this effect.

For both sampling error and within-species variation, contrasts are standardized to give them a common variance, which may cause a heavy exaggeration for close relatives after conversion. This is true because they may largely have arisen through sampling error or within-species variation, rather than through evolution [68]. These issues can be solved differently: Methods can be employed that explicitly consider this within-species variation while using phylogenetically independent contrasts [20, 36]. Another option is to apply a branch length transformation for the

given phylogeny. Branch lengths are assumed to be proportional to expected variance of character evolution for the trait(s) of interest. In general, such branch lengths cannot be known directly, and testing phylogenetic signals in comparative approaches often requires appropriate branch lengths [7]. Therefore, reasonable approximations must be employed. The issue of branch length error on the performance of phylogenetically independent contrasts has been intensively discussed (e.g., see [14] for details), and several transformations have been proposed. Some of them are already implemented in phylogeny programs (e.g., Mesquite [45]), but a detailed review of the different transformations is beyond the scope of this thesis[1] (e.g., equal branch lengths (see [49]), Grafen's rho method [26], Grafen's arbitrary method [26], Pagel's method [57], and Nee's arbitrary method (cited in [62, p. 416]). Nevertheless, it can reduce type 1 errors and the artifacts described above, particularly by extending branches close to the tips using the *rho transformation*. Rho values smaller than 1 compress the tree near the root and expand it near the tips, whereas values bigger than 1 compress the tree near the tips and expand it near the root [14]. Thus, in our case, rho values smaller than 1 are worth applying.

To summarize, the presented biases can overestimate the importance of certain species pairs if they are close relatives, and careful consideration is mandatory.

7.1.2 Phylogenetic Errors

With most phylogenetic methods, the given phylogeny is assumed to be completely true and without any error. Clearly, this is very unrealistic, and in reality, we are never able to check the validity of this assumption. Errors can manifest themselves in different ways: Branch lengths can be estimated imprecisely or topological error can occur. Generally speaking, both types of error are likely to be important. Simulation studies for the method of phylogenetically independent contrasts method exist, and they show that the method is relatively robust against such biases [48]. However, it could lead to the identification of key species that are only selected because of those inaccuracies.

To control for uncertainty in phylogenetic relationships, we can use Bayesian approaches to estimate multiple phylogenies not contingent on any single phylogeny or set of branch lengths [56, 35].

7.1.3 Statistical Power and Independency of Data Points

Another important point is that the pairwise comparisons method has less statistical power than the original phylogenetically independent contrasts method. Due to the fact that it compares only the tips of the phylogeny, and does not calculate comparisons based on inferred values at interior nodes, fewer comparisons can be made. That is, it loses information in focusing only on a subset of branches and comparisons.

At least six pairwise comparisons are needed to demonstrate statistical significance in a non-parametric test. Nevertheless, the more comparisons that are available, the higher the power to detect differences, and the easier it is to control for other variables that also might have an influence.

[1] e.g., see the PDAP [51] manual for overview (www.biology.ucr.edu/people/faculty/Garland/PDTREE_Mesquite.doc)

An area for future research is to explore how using phylogenetic targeting might improve the performance of the more general method of independent contrasts, in which contrasts are calculated throughout the phylogeny. Sister species selected in phylogenetic targeting will be used also in the calculation of independent contrasts, and thus should add to the power of this method. It seems likely that the advantages will translate to the more general method, with specific advantages depending on the distribution of traits across the entire tree.

7.1.4 Non-congeneric Species Pairs

A major advantage of the pairwise comparison approach is that few confounding variables are expected to influence the results if sister species pairs are considered [52, 33]. In our approach, however, all possible pairwise comparisons are generated. More distantly related species pairs can also be selected, e.g. those that use more than two branches. That can be critical, because there may be other, non-examined confounding variables that are the true reason for the difference. The more distantly related two species are, the more likely it is that such an effect could bias the result.

However, this is usually controlled in practice, because the contrast standardization (if enabled) strongly weakens the score of a distantly related species pair. In the maximal pairing algorithm, such a pair would be selected only in cases where either the tree topology is the reason (e.g., only one free path is available and all other species have already been paired) or that particular pair has, depending on the setup, extreme differences in the hypotheses. More generally, by requiring that the alternative hypotheses have data on variables of interest to investigate those hypotheses, it should also be possible to statistically control for confounds among the species when conducting the actual comparative test of the adaptive hypothesis. Nonetheless, it is important to emphasize that these pairs have to be critically considered in light of possible confounds.

7.1.5 Discrete Data Character States

Discrete characters have, by definition, only a finite number of possible states. They can be treated as ordered or unordered (see [76] for an overview, see also Figure 7.1). For binary characters (two states), this makes no difference; for a multi-state character (more than two states), however, ordered means that they have a particular sequence in which the states must occur through evolution. Intermediate states are involved, and the costs between different pairs of states are different.

Treating a character as unordered means that every state change is equal. They do not require passing through intermediate states, because each state can directly be transformed into any other state.

The *PhyloTargeting* application treats all discrete characters as unordered. Should a user have ordered characters to analyze, a temporary work-around is to enter that data in a continuous data table in Mesquite, which would thus treat larger changes (more discrete steps) as more important than smaller changes. Future work could implement analyses for ordered character series in the program, if demand for this feature arises.

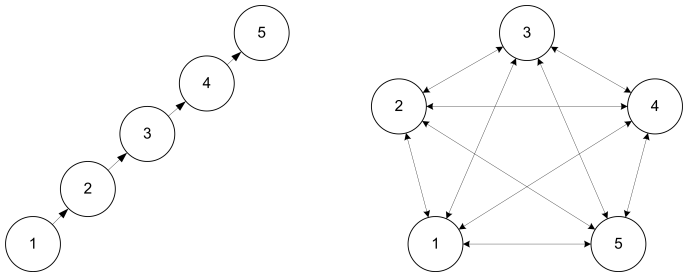


FIGURE 7.1: A character can be either categorized as ordered, represented in a transformation series (left), or as unordered, represented as a complete graph (right). In the former case, the sequence of states is fixed: To reach state 5 from state 2 one must go through states 3 and 4. Moreover, character changes from state 1 to 2 have to be treated differently than changes from state 1 to 3. The figure has been modified from [29].

7.1.6 Polytomies

Another issue that arises is that of polytomies in the tree. When conducting comparative studies, one will often be faced with polytomous phylogenies. Polytomies can affect the conclusions we draw, because most methods are sensitive to them and it has been studied intensively in the comparative context [65, 57, 58]. Due to the dubious nature of polytomies, hard as well as soft polytomies have to be considered separately.

Hard polytomies are usually represented as a series of bifurcations with branch lengths of 0, and the branches are joined arbitrarily. With the original phylogenetically independent contrasts method, this has no effect on the calculations [23]. However, in our approach, it indeed makes a difference if the polytomy is represented as series of 0-bifurcations or as true polytomy. In the former case, less pairs can be selected. This is due to the fact that we use pairwise comparisons, and we do not allow for any branch to be allocated to more than one pairwise comparison. Another reason for this discrimination is that the maximal pairing algorithm is much faster for dichotomous nodes, and sometimes not feasible for trees with both a high number of taxa and a large degree of polytomies. Thus, by introducing zero branch lengths, a faster execution time can be achieved. Although this procedure loses some information, it may be sometimes preferable.

For soft polytomies, five different methods have been proposed for the original phylogenetically independent contrasts method (see [22] for an overview). Some of them involve the use of simulated phylogenies [41] or random trees [47], others adjust the number of contrasts that are computed or the degrees of freedom [66, 59, 60, 26, 27]. In this version of the program, we treat soft and hard polytomies the same. This may be further extended by using one or more of the existing approaches developed in previous work.

A last issue that arises with polytomies is the calculation of the maximal pairing. For polytomous nodes, it has an execution time that is exponential to the degree of the polytomy for a node. Thus, for phylogenies with a high degree of polytomies, a noticeable delay emerges. Although faster algorithms exist, it is not implemented for several reasons. If the algorithm takes too long in a particular case, then it might be worthwhile to transform all polytomies to zero branch lengths. However, if no node has a degree bigger than ten, the algorithm should still be fast enough to work well in practice.

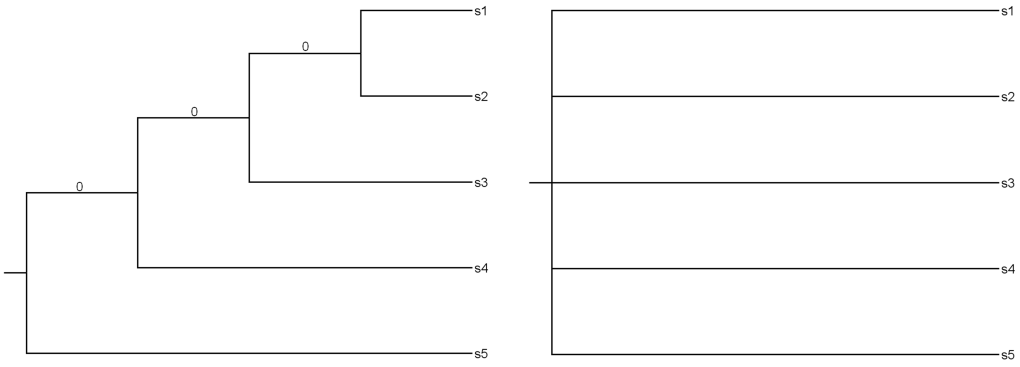


FIGURE 7.2: An example phylogeny that contains a polytomy, shown in two different ways. On the left, the polytomy is represented as a series of random zero branches. On the right, it is represented without zero branches. See text for details.

7.2 Directions for Future Research

The presented methodology is only an initial framework, and more sophisticated methods may evolve in the future. We now highlight some additional extensions and generalizations of the methodology itself:

- The scoring system and standardization procedure can be modified. Standardization can also be applied before the values are transformed, rather than afterwards. Moreover, more sophisticated scoring mechanisms could be applied. The developed scoring system does not have an explicit theoretical foundation; instead, it is chosen to address a practical problem.
- It is imaginable to not only score pairwise comparisons, but also to score whole lineages (e.g., a lineage in the tree that represents all new-world monkeys). This can be helpful to identify subgroups that are in general more interesting in relation to a particular problem.

We also want to separately emphasize possible modifications to the algorithm that determines the maximal pairing, because this idea is central to the method, and adaptations to this algorithm could extend the usefulness and application of the approach:

- Instead of determining only the maximal pairing, the algorithm could be extended to find also suboptimal pairings or, more generally, pairings with a score above a given threshold. This can be achieved by using a variant of the original dynamic programming approach and the backtracking procedure.
- Another development of possible interest would involve developing the program such that users select favored comparisons or species, and the algorithm returns the maximal pairing that includes these comparisons or species.
- Lastly, one could imagine taking into account the costs of collecting additional data, including prior knowledge about the costs of collecting data for particular species (e.g., these costs can be high if the species is rare). Within a particular budget, which phylogenetically independent measures should then be taken? Which species are the most compelling, given the limited resources researchers often have?

7.3 Conclusion

In summary, we created a new, quantitative framework that is able to address many different questions related to data collection in the context of finite resources. It can be therefore beneficial for a variety of people, and not only for comparative biologists, because the general approach can easily be extended to other areas. Moreover, the developed algorithms may be applied to other fields to solve optimization problems based on phylogenies. From an evolutionary prospective, this new methodology may also help to give new insights why species manifested such substantial diversity.

Appendix A

Installation of the *PhyloTargeting* Program (Downloadable Version)

To run the program locally, or on your own server, simply follow the instructions below.

Requirements:

- Web server with support for *.htaccess* overwritable MIME types (e.g., Apache[1]). If you do not have any experience in installing a web server and its configuration, we recommend the use of XAMPP[2] (available for Windows, Linux and Mac OS X) or a similar package for a web server, because the installation is extremely easy (just unpack the archive, with no further installation necessary) and it does not influence the system. Be aware that the security settings are very generous in the default setting, and modify these settings to ensure better security.
- PHP Version 5 (tested with version 5.2.3)
- Some PHP5 libraries may be installed to enable all features, e.g. the GD library for tree drawing (in XAMPP, this is enabled by default).
- A modern browser: *PhyloTargeting* works with the following browsers: *Firefox*, *Mozilla*, *Internet Explorer*, *Opera*, *Safari*. However, there might be small layout differences with different browsers. This is due to the fact there is no common accepted standard yet. The application has been developed in *Firefox*, so this might be the best choice.

Installation

1. All required files can be downloaded as an archive from the website of the *PhyloTargeting* program. This archive contains the following folders and files:
   - *main- folder*: contains the HTML and PHP files that are needed for the interface
   - *src- folder*: contains necessary basic files and all class files
   - *pic- folder*: contains images that appear on the website
   - *lib- folder*: contains external libraries (PEAR:HTTP_Upload[3], fpdf[4], PHP progressbar [8])

[1] http://httpd.apache.org
[2] http://www.apachefriends.org
[3] http://pear.php.net/package/HTTP_Upload
[4] http://www.fpdf.org/

- *doc- folder*: contains the documentation and example files
- *css- folder*: contains the css file

2. Unpack the archive into a suitable directory on your web server. Be sure that this directory is set to allow configuration overwrites using *.htaccess*. Furthermore, make sure that the directory is writable by the web user.
3. Put the *PhyloTargeting* folder in the correct directory of your web server (e.g., in the htdocs/xampp directory if you use XAMPP).
4. Open a web browser and type : http://127.0.0.1//*webserver_specific_path*/PhyloTargeting, where *webserver_specific_path* reflects the path directory to the *PhyloTargeting* folder.
5. You should now be able to see the *index.html* page of the program.

Settings

A few settings that can be adjusted if necessary. Specifically, three options can be changed in the *GeneralSettings* file in the *src-folder*:

1. *MaxCombs*: The maximum number of pairwise comparisons that are allowed.
2. *MaxTime*: The maximal number of seconds a script is allowed to run[5].
3. *MaxMemory*: The maximal number of memory (in megabytes) that can be allocated from a script. If the maximal memory allocation is set too low, then execution of memory-consuming scripts will stop due to an *Out-of-Memory* exception. The value should be set to values above 100-150 MB to guarantee compatibility with large datasets.

A modification of these settings is trivial: One only has to change these values in the *GeneralSettings* file. *MaxTime* and *MaxMemory* will be updated immediately for every session and without any consequences. However, the updated *MaxCombs* value will not take effect before a new data file is submitted. Thus, already existing session files will still use the old setting.

Support

If any problems are encountered (bugs, difficulties in installing the software), do not hesitate to contact the author (*chrarnold (at) web.de*).

[5] see http://de3.php.net/set_time_limit for details

Appendix B

Specification of the Supported NEXUS Elements

PhyloTargeting is able to read NEXUS files that have been created with the software Mesquite [45]; NEXUS files from other software might be compatible as well. *PhyloTargeting* will encounter an error and stop immediately if any incompatibility is detected.

```
#NEXUS
[written Sat Apr 19 15:49:37 CEST 2008 by Mesquite  version 2.0 (build i69) at CHRIS2/192.168.2.100]

BEGIN TAXA;
        TITLE Phylocom_Phylogeny_Taxa;
        DIMENSIONS NTAX=3;
        TAXLABELS
                Allocebus_trichotis Arctocebus_calabarensis Avahi_laniger
        ;
END;

BEGIN CHARACTERS;
        TITLE  continuous_trait_data;
        DIMENSIONS  NCHAR=3;
        FORMAT DATATYPE = CONTINUOUS;
CHARSTATELABELS
                1 Group_Size,
                2 Home_Range,
                3 Longevity ;
        MATRIX
        Allocebus_trichotis       4.0 ? ?
        Arctocebus_calabarensis   1.5 ? 13.0
        Avahi_laniger             2.5 2.0 ?
;
END;
BEGIN CHARACTERS;
        TITLE  discrete_trait_data;
        DIMENSIONS  NCHAR=2;
        FORMAT DATATYPE = STANDARD GAP = - MISSING = ? SYMBOLS = "  0 1 ";
        CHARSTATELABELS
                1 CognitiveStudy, 2 ActivityPeriod ;
        MATRIX
        Allocebus_trichotis      01
        Arctocebus_calabarensis  01
        Avahi_laniger            10
;

END;
BEGIN TREES;
        Title 'Trees from "primates_list3.nex"';
        LINK Taxa = Phylocom_Phylogeny_Taxa;
        TRANSLATE
                1 Allocebus_trichotis,
                2 Arctocebus_calabarensis,
                3 Avahi_laniger;
        TREE 'mammals primates upperDates++' = ((1:55.10000000000001,3:55.100001000000006):24.5,2:79.6):10.8;
        TREE 'mammals primates2 lowerDates++' = ((1:43.099999,3:43.100001):28.200001,2:71.300002):13.8;
END;
```

FIGURE B.1: A simple example file consisting of three species. It was created with Mesquite and has a full compatibility to the *PhyloTargeting* application.

The following specification presents mandatory elements as well as recommended optional elements; all other present blocks are simply ignored. Thus, in general, NEXUS files do not have to be preprocessed to be compatible with *PhyloTargeting*, as long as they contain the necessary standard blocks together with valid information in it.

Every NEXUS file must also start with a `#NEXUS` line. Furthermore, in the original NEXUS specification [42], eight primary public blocks are described and mentioned. However, we support only the following 3 blocks: `TAXA`, `CHARACTERS`, and `TREES`. All of them are obligatory and each block has to start with a `'BEGIN [name of block]'` statement and has to end with an `'END;'` statement. These two statements are not allowed to appear within an already started block (see NEXUS specification for details).

We now describe mandatory and optional elements within these blocks and all optional statements are marked with `*`.

- `CHARACTERS` block

  This block defines characters or traits that represent biological features. It must appear at least once, but several instances are supported and common. However, we have to distinguish between continuous characters and discrete characters, and describe both kinds separately. Character values can be missing, and this should be indicated by a `?` symbol. Characters can be labeled, but this feature is optional. If no labeling is found, then the name of the trait will be `undef` in the *PhyloTargeting* application.

  1. Continuous characters

     At least one blank is mandatory between the character values, and delimiter symbol must be `'.'`.

```
BEGIN CHARACTERS;
DIMENSIONS NCHAR= [number_of_characters];
FORMAT DATATYPE = CONTINUOUS
CHARSTATELABELS*
 1 [char_name_1],
 2 [char_name_2],
 ...
 m [char_name_m];
MATRIX
 [species_name_1]  [value_char_1]  [value_char_2] ... [value_char_m]
 [species_name_2]  [value_char_1]  [value_char_2] ... [value_char_m]
 ...
 [species_name_n]  [value_char_1]  [value_char_2] ... [value_char_m]
;
END;
```

  2. Discrete characters

     The default symbol for missing values is `?`, and character values must not be separated by blanks.

```
BEGIN CHARACTERS;
```

```
DIMENSIONS NCHAR= [number_of_characters];
FORMAT DATATYPE = STANDARD MISSING = [symbol_for_missing_data]
 SYMBOLS = [all_valid_states]
CHARSTATELABELS*
 1 [char_name_1],
 2 [char_name_2],
 ...
 m [char_name_m];
MATRIX
 [species_name_1][value_char_1][value_char_2]...[value_char_m]
 [species_name_2][value_char_1][value_char_2]...[value_char_m]
 ...
 [species_name_n][value_char_1][value_char_2]...[value_char_m]
;
END;
```

- `TAXA` block

  This block defines the taxa that are used in the data file. The block must appear exactly once, and the number of `NTAX` must be identical with the number of specified taxa in the `TAXLABELS` statement. The format should be as follows:

```
BEGIN TAXA;
DIMENSIONS NTAX=[number_of_species];
TAXLABELS [species_name_1] [ species_name_2]...[ species_name_n];
END;
```

- `TREE` block

  This block must appear exactly once and defines an arbitrary number of trees on which the calculation is based. At least one tree must be specified, but more than one is possible as well. All trees must be in valid NEWICK[1] format.

```
BEGIN TREES;
TRANSLATE
 [1] [species_name_1],
 [2] [species_name_2],
 ...
 [n] [species_name_n];
TREE [tree_name_1] = [tree_1_in_NEWICK_format];
TREE [tree_name_2] = [tree_2_in_NEWICK_format];
...
TREE [tree_name_k] = [tree_k_in_NEWICK_format];
END;
```

[1] http://evolution.genetics.washington.edu/phylip/newicktree.html

Appendix C

Description of Comparative Analysis Procedures for Creating the Sleep Dataset

In the following, we describe the procedure for estimating the residuals between brain mass and body mass using phylogenetically independent contrasts. This was the final step for generating the sleep dataset, and requires some explanation.

1. We expect a highly significant correlation between body mass and brain mass [32]. We therefore have to control for this effect, which we did by taking the residuals from the regression line [12]. With non-phylogenetic analysis, the estimated slope is potentially biased (see [18]). Thus, we need to estimate the slope while considering the phylogeny.
2. As with all allometric analyses, we $\log_{10}$-transformed the original brain mass and body mass values for the phylogenetic independent contrasts analysis. We then regressed log-transformed values of brain mass on log transformed values of body mass using independent contrasts, as described in steps 3-4, using the PDAP module of Mesquite [51].
3. Determine if the assumptions of phylogenetic independent contrasts are met:
   - Check the absolute values of the standardized phylogenetic independent contrasts versus their standard deviations, represented as the square root of the sum of the branch lengths [49, 23]. This diagnostic is the most commonly used check of whether the data and branch lengths meet the assumptions of Brownian motion evolution (e.g., see [14, 13]). There should be no significant correlation (flat regression line). However, if a significant correlation is found, one can conclude that the data and phylogeny do not meet the assumptions. As brain mass failed to meet this assumption at P = 0.01, we applied a branch length transformation within Mesquite (Rho transform [26] with rho = 0.5). After the branch transformation, the two-tailed P-values for the log-transformed traits were 0.44 and 0.67.
   - Other, less studied assumptions are described in the PDAP manual[1].
4. After applying the rho transformation, we determined the slope using least square regression (b = 0.69, R^2 = 0.86), which was computed through the origin [23].
5. Reconstruct the root node values using PDAP (brain mass = 2.83, body mass = 6.82). The justification is that the regression line must go through the mean in the raw data space,

[1] http://www.biology.ucr.edu/people/faculty/Garland/PDTREE_Mesquite.doc

and the root node reconstruction is exactly the mean of the whole tree [24]. Thus, one can estimate the y-intercept of the regression line (-1.91).

6. Estimate the residuals using the new regression line by calculating the difference between the expected values and the observed values. These differences define our new trait – residual brain mass – that is used in the analysis. A positive residual reflects that for its body mass, a species has a larger than expected brain mass, while a negative residual reflects the opposite.

Bibliography

[1] T. Allison and D. V. Cicchetti. Sleep in mammals: ecological and constitutional correlates. *Science*, 194:732–734, 1976.

[2] M. A. Bender and M. Farach-Colton. The LCA problem revisited. In *Proc. 4th Latin American Symposium on Theoretical Informatics, volume 1776 of LNCS*, pages 88–94. Springer, 2000.

[3] F. G. Benedict. *Vital energetics.* Carnegie Institute Publications, 503. Washington, D. C., 1938.

[4] O. R. P. Bininda-Emonds, M. Cardillo, K. E. Jones, R. D. E. MacPhee, R. M. D. Beck, R. Grenyer, S. A. Price, R. A. Vos, J. L. Gittleman, and A. Purvis. The delayed rise of present-day mammals. *Nature*, 446:507–512, 2007.

[5] T. R. Birkhead and J. D. Biggins. Reproductive synchrony and extra-pair copulations in birds. *Ethology*, 74:320–334, 1987.

[6] T. Blackburn. The interspecific relationship between egg size and clutch size in wildfowl. *Auk*, 108:209–211, 1990.

[7] S. P. Blomberg, T. Garland Jr., and A. R. Ives. Testing for phylogenetic signal in comparative data. *Evolution*, 57:717–745, 2003.

[8] D. Bongard. PHP-Progressbar - PHP-Klasse für Fortschrittsbalken. `http://www.bongard.net/blog/2007/04/18/php-progressbar_fortschrittsbalken/`, 2007. Available online, last visited on April 3rd 2008.

[9] I. Capellini, R. A. Barton, P. McNamara, B. T. Preston, and C. L. Nunn. Phylogenetic analysis of the evology and evolution of mammalian sleep. *Evolution*, 2008. doi:10.1111/j.1558-5646.2008.00392.x, in press.

[10] P. Carvalho, A. F. Diniz-Filho, and L. M. Bini. The impact of Felsenstein's "Phylogenies and the comparative method" on evolutionary biology. *Scientometrics*, 62:53–66, 2005.

[11] C. Chambers, J. Dolske, and J. Iyer. TCP/IP Security. `http://www.linuxsecurity.com/resource_files/documentation/tcpip-security.html`. Available online, last visited on November 7th 2007.

[12] R. O. Deaner, C. L. Nunn, and C. P. v. Schaik. Comparative tests of primate cognition: different scaling methods produce different results. *Brain, Behavior, and Evolution*, 55:44–52, 2000.

[13] R. Díaz-Uriarte and T. Garland Jr. Testing hypotheses of correlated evolution using phylogenetically independent contrasts: sensitivity to deviations from Brownian motion. *Systematic Biology*, 45:27–47, 1996.

[14] R. Díaz-Uriarte and T. Garland Jr. Effects of branch length errors on the performance of phylogenetically independent contrasts. *Systematic Biology*, 47:654–72, 1998.

[15] M. A. Elgar, M. D. Pagel, and P. H. Harvey. Sleep in mammals. *Animal Behavior*, 36:1407–1419, 1988.

[16] S. Esser. $GLOBALS Overwrite and it's Consequences. `http://www.hardened-php.net/index.76.html`, 2005. Available online, last visited on November 7th 2007.

[17] D. P. Faith. Phylogenetic pattern and the quantification of organismal biodiversity. *Philosophical Transactions of the Royal Society B*, 345:45–58, 1994.

[18] J. Felsenstein. Phylogenies and the Comparative Method. *The American Naturalist*, 125:1–15, 1985.

[19] J. Felsenstein. Phylogenies from molecular sequences: inference and reliability. *Annual review of genetics*, 22:521–565, 1988.

[20] J. Felsenstein. Comparative methods with sampling error and within-species variation: contrasts revisited and revised. *The American Naturalist*, 171:713–25, 2008.

[21] H. N. Gabow. An Efficient Implementation of Edmonds' Algorithm for Maximum Matching on Graphs. *Journal of the Association for Computing Machinery*, 23:221–234, 1976.

[22] T. Garland Jr. and R. Díaz-Uriarte. Polytomies and Phylogenetically Independent Contrasts: Examination of the Bounded Degrees of Freedom Approach. *Systematic Biology*, 48:547Ű–558, 1999.

[23] T. Garland Jr., P. H. Harvey, and A. R. Ives. Procedures for the analysis of comparative data using phylogenetically independent contrasts. *Systematic Biology*, 41:18–32, 1992.

[24] T. Garland Jr. and A. R. Ives. Using the past to predict the present: Confidence intervals for regression equations in phylogenetic comparative methods. *The American Naturalist*, 155:346–364, 2000.

[25] T. Garland Jr., P. E. Midford, and A. R. Ives. An Introduction to Phylogenetically Based Statistical Methods, with a New Method for Confidence Intervals on Ancestral Values. *American Zoologist*, 39:374–388, 1999.

[26] A. Grafen. The phylogenetic regression. *Philosophical Transactions of the Royal Society B*, 326:119–157, 1989.

[27] A. Grafen. The uniqueness of the phylogenetic regression. *Journal of Theoretical Biology*, 156:405–423, 1992.

[28] D. Gusfield. *Algorithms on strings, trees, and sequences: computer science and computational biology*. Cambridge University Press, New York, 1997.

[29] G. Hagedorn. Deltaaccess: 'Describe' & 'Identify'- A SQL interface to DELTA (Description Language for Taxonomy), implemented in Microsoft Access, user guide and documentation. Technical report, 1999. Available online, last visited on May 23th 2008.

[30] W. D. Hamilton and M. Zuk. Heritable true fitness and bright birds: a role for parasites? *Science*, 218:384–387, 1982.

[31] L. J. Harmon and J. B. Losos. The effect of intraspecific sample size on type I and type II error rates in comparative studies. *Evolution*, 59:2705–2710, 2005.

[32] P. H. Harvey and J. R. Krebs. Comparing brains. *Science*, 249:140–146, 1990.

[33] P. H. Harvey and M. D. Pagel. *The Comparative Method in Evolutionary Biology*. Oxford Univ. Press, Oxford, 1991.

[34] Hermelin, D. Constant-Time LCA Retrieval. `http://cs.haifa.ac.il/LANDAU/courses/String/String_files/LCA.ppt`, 2005. Available online, last visited on June 8th 2008.

[35] J. P. Huelsenbeck, B. Rannala, and J. P. Masly. Accommodating phylogenetic uncertainty in evolutionary studies. *Science*, 288:2349–2350, 2000.

[36] A. R. Ives, P. E. Midford, and T. Garland Jr. Within-species variation and measurement error in phylogenetic comparative methods. *Systematic Biology*, 56:252–270, 2007.

[37] M. Kleiber. Body size and metabolism. *Hilgardia*, 6:315–353, 1932.

[38] Koskela, M. Normalizing and Redistributing Variables. `www.cis.hut.fi/Opinnot/T-61.6010/s99/presentations/oct27_MK.ppt`, 1999. Available online, last visited on June 8th 2008.

[39] J. A. Lesku, T. C. Roth, C. J. Amlaner, and S. L. Lima. A phylogenetic analysis of sleep architecture in mammals: the integration of anatomy, physiology, and ecology. *The American Naturalist*, 168:441–453, 2006.

[40] S. L. Lima, N. C. Rattenborg, J. A. Lesku, and C. J. Amlaner. Sleeping under the risk of predation. *Animal Behavior*, 70:723–736, 2005.

[41] J. B. Losos. An approach to the analysis of comparative data when a phylogeny is unavailable or incomplete. *Systematic Biology*, 43:117–123, 1994.

[42] D. R. Maddison, D. L. Swofford, and W. P. Maddison. NEXUS: An Extensible File Format for Systematic Information. *Systematic Biology*, 46:590–621, 1997.

[43] W. P. Maddison. A Method for Testing the Correlated Evolution of Two Binary Characters: Are Gains or Losses Concentrated on Certain Branches of a Phylogenetic Tree? *Evolution*, 44:539–557, 1990.

[44] W. P. Maddison. Testing Character Correlation using Pairwise Comparisons on a Phylogeny. *Journal of Theoretical Biology*, 202:195–204, 2000.

[45] W. P. Maddison and D. R. Maddison. Mesquite: A modular system for evolutionary analysis. `http://mesquiteproject.org`, 2007. Version 2.0.

[46] March Browsers Market Share Results. `http://www.favbrowser.com/category/market-share/`, 2008. Available online, last visited on April 9th 2008.

[47] E. P. Martins. Conducting phylogenetic comparative studies when the phylogeny is not known. *Evolution*, 50:12–22, 1996.

[48] E. P. Martins, J. A. F. Diniz-Filho, and E. A. Housworth. Adaptive constraints and the phylogenetic comparative method: a computer simulation test. *Evolution*, 56:1–13, 2002.

[49] E. P. Martins and T. Garland Jr. Phylogenetic analyses of the correlated evolution of continuous characters: a simulation study. *Systematic Biology*, 45:534–557, 1991.

[50] P. McNamara, I. Capellini, E. Harris, C. L. Nunn, R. A. Barton, and B. Preston. The phylogeny of sleep database: A new resource for sleep Scientists. *The Open Sleep Journal*, 1:11–14, 2008.

[51] P. E. Midford, T. Garland Jr., and W. P. Maddison. Pdap package of mesquite, 2005.

[52] A. P. Møller and T. R. Birkhead. A Pairwise Comparative Method as Illustrated by Copulation Frequency in Birds. *The American Naturalist*, 139:644–656, 1992.

[53] C. L. Nunn. Comparative Primatology: The Ecology and Evolution of Primate Sleep. `http://www.leipzig-school.eva.mpg.de/files/sleep.htm`. Available online, last visited on April 22th 2008.

[54] C. L. Nunn, P. McNamara, I. Capellini, B. Preston, and R. A. Barton. Primate sleep in phylogenetic perspective. In P. McNamara, R. A. Barton, and C. L. Nunn, editors, *The evolution of sleep*. Cambridge University Press. in press.

[55] C. L. Nunn and C. P. v. Schaik. Reconstructing the behavioral ecology of extinct primates. In J. M. Plavcan, R. F. Kay, W. L. Jungers, and C. P. v. Schaik, editors, *Reconstructing Behavior in the Fossil Record*, pages 159–216. Kluwer Academic / Plenum, New York, 2002.

[56] M. Pagel and F. Lutzoni. Accounting for phylogenetic uncertainty in comparative studies of evolution and adaptation. In M. Lässig and A. Valleriani, editors, *Biological Evolution and Statistical Physics*, pages 148–161. Springer-Verlag, Berlin, 2002.

[57] M. D. Pagel. A method for the analysis of comparative data. *Journal of Theoretical Biology*, 156:431–442, 1992.

[58] M. D. Pagel. Seeking the evolutionary regression coefficient: an analysis of what comparative methods measure. *Journal of Theoretical Biology*, 164:191–205, 1993.

[59] M. D. Pagel and P. H. Harvey. Comparative methods for examining adaptation depend on evolutionary models. *Folia Primatologica*, 53:203–220, 1989.

[60] M. D. Pagel and P. H. Harvey. On solving the correct problem: Wishing does not make it so. *Journal of Theoretical Biology*, 156:425–430, 1992.

[61] D. E. L. Promislow. New perspectives on comparative tests of antagonistic pleiotropy using Drosophila. *Evolution*, 49:394–397, 1995.

[62] A. Purvis. A composite estimate of primate phylogeny. *Philosophical Transactions of the Royal Society B*, 348:405–421, 1995.

[63] A. Purvis, P. M. Agapow, J. L. Gittleman, and G. M. Mace. Nonrandom extinction and the loss of evolutionary history. *Science*, 288:328–330, 2000.

[64] A. Purvis and L. D. Bomham. Estimating the transition/transversion ratio from independent pairwise comparisons with an assumed phylogeny. *Journal of Molecular Evolution*, 44:112–119, 1997.

[65] A. Purvis and T. Garland Jr. Polytomies in comparative analyses of continuous characters. *Systematic Biology*, 42:569–575, 1993.

[66] A. Purvis and T. Garland Jr. Polytomies in comparative analyses of continuous data. *Systematic Biology*, 42:569–575, 1993.

[67] A. Purvis and A. Rambaut. Comparative analysis by independent contrasts (CAIC): an Apple Macintosh application for analysing comparative data. *Computer Applications in the Biosciences*, 11:247–251, 1995.

[68] A. Purvis and A. J. Webster. Phylogenetically independent comparisons and primate phylogeny. In P. C. Lee, editor, *Comparative Primate Socioecology*, pages 44–70. Cambridge University Press, Cambridge, 1999.

[69] A. F. Read and P. H. Harvey. Reassessment of comparative evidence for Hamilton and Zuk theory on the evolution of secondary sexual characters. *Nature*, 339:618–620, 1989.

[70] A. F. Read and S. Nee. Inference from binary comparative data. *Journal of Theoretical Biology*, 173:99–108, 1995.

[71] S. M. Reader and L. K. N. Social intelligence, innovation, and enhanced brain size in primates. *Proceedings of the National Academy of Sciences of the United States of America*, 99:4436–4441, 2002.

[72] R. E. Ricklefs and J. M. Starck. Applications of phylogenetically independent contrasts: A mixed progress report. *Oikos*, 77:167–172, 1996.

[73] E. J. Salisbury. *The Reproductive Capacity of Plants*. G. Bell & Son, London, 1942.

[74] G. Schlossnagle. *Advanced PHP Programming*. Sams, 2004.

[75] J. M. Siegel. Phylogeny and the function of REM sleep. *Behavioural Brain Research*, 69:29–34, 1995.

[76] J. B. Slowinski. "Unordered" Versus "Ordered" Characters. *Systematic Biology*, 42:155–165, 1993.

[77] J. R. Stevens, J. N. Wood, and M. D. Hauser. When quantity trumps number: discrimination experiments in cotton-top tamarins (Saguinus oedipus) and common marmosets (Callithrix jacchus). *Animal Cognition*, 10:429–437, 2007.

[78] Trenham, P. C. et. al. Primenet, Ultraviolet Radiation /Amphibian Populations, Research Planning Workshop. `http://www.forestry.umt.edu/research/MFCES/programs/primenet/Assets/UV%%20Amphibians/amphib.pdf`, 1999. Available online, last visited on June 8th 2008.

[79] U.S. GLOBEC. Dynamics of Open Ocean Populations - Report of a U.S. GLOBEC Workshop, U.S. Global Ocean Ecosystems Dynamics, Report Number 14. `http://www.usglobec.org/reports/pdf/rep14.pdf`, 1995. Available online, last visited on June 8th 2008.

Affirmation

Hereby I explain to have written this work independently and only to have used the sources and aids stated in the bibliography.

Place and date DD/MM/YYYY	Signature

Erklärung

Ich versichere, dass ich die vorliegende Arbeit selbständig und nur unter Verwendung der angegebenen Quellen und Hilfsmittel angefertigt habe, insbesondere sind wörtliche oder sinngemäße Zitate als solche gekennzeichnet. Mir ist bekannt, dass Zuwiderhandlung auch nachträglich zur Aberkennung des Abschlusses führen kann.

Ort und Datum	Unterschrift