HELSINKI UNIVERSITY OF TECHNOLOGY
Department of Computer Science and Engineering
Degree programme of Information Networks

Hannes Heikinheimo

Inferring taxonomic hierarchies from 0-1 data

Master's Thesis submitted in partial fulfillment of the requirements for the degree of Master of Science in Technology.

Espoo, September 30, 2005

Supervisor: Professor Heikki Mannila
Instructor: Professor Heikki Mannila

HELSINKI UNIVERSITY OF TECHNOLOGY

ABSTRACT OF THE MASTER'S THESIS

Author:	Hannes Heikinheimo
Name of the Thesis:	Inferring taxonomic hierarchies from 0-1 data
Date:	September 30, 2005 **Number of pages:** 59
Department:	Department of Computer Science and Engineering
Professorship:	T-93 Knowledge Engineering
Supervisor:	Prof. Heikki Mannila
Instructor:	Prof. Heikki Mannila

A taxonomic hierarchy is a classification of objects into a hierarchy of categories organized by a set of subclass relations. Taxonomic hierarchies are widely used to model data both in scientific and business related domains. Examples of applications can be found among others from systematic biology, medicine, market research and artificial intelligence.

This thesis is concerned with the inference of taxonomic hierarchies. The work is divided into two parts. In the first part, methods and theoretical considerations are discussed. The focus is on techniques that can be used to generate and compare taxonomic hierarchies. Also, definitions of dissimilarity between data objects are an important theme. In the second part of the thesis, the discussed techniques and definitions are applied to a data set of occurrence range information of European land mammals. The practical research problem of the thesis is to find out whether the data supports a taxonomic hierarchy of mammal occurrence.

The behavior of several occurrence based distance measures for the mammals is analyzed. Furthermore, a set of taxonomic hierarchies is generated using agglomerative clustering and a greedy hierarchy tree searching strategy. The fit of the data to the taxonomic hierarchies is assessed with two resampling methods: the Monte Carlo method and the Bootstrap method. The results suggest a credible geographical hierarchy of mammal species in Europe.

Keywords: hierarchy, taxonomy, ultrametric, binary tree, distance measure, clustering, tree comparison, model validity

TEKNILLINEN KORKEAKOULU

DIPLOMITYÖN TIIVISTELMÄ

Tekijä: Hannes Heikinheimo

Työn nimi: Hierarkkisten luokittelujen päättely 0-1 aineistosta

Päivämäärä: 30.9.2005 **Sivuja:** 59

Osasto: Tietotekniikan osasto

Professuuri: T-93 Tietämystekniikka

Työn valvoja: Prof. Heikki Mannila

Työn ohjaaja: Prof. Heikki Mannila

Hierarkkinen luokittelu on oliojoukon lajittelu hierarkkisesti organisoituihin kategorioihin ja näiden alikategorioihin. Hierarkkinen luokittelu on paljon käytetty tekniikka niin tieteellisen kuin kaupallisenkin tiedon mallintamisessa. Esimerkkejä sovellusalueista löytyy muun muassa systeemibiologian, lääketieteen, asiakasdata-analyysin ja tekoälyn piiristä.

Tässä diplomityössä käsitellään hierarkkisten luokittelujen päättelyä ja siihen liittyviä kysymyksiä. Työn rakenteen voi jakaa kahteen osaan. Ensimmäisessä osassa tarkastellaan hierarkkisten luokittelujen päättelymenetelmiä sekä niihin liittyvää teoriaa. Erityisesti diplomityö keskittyy joihinkin hierarkkisten luokittelujen muodostusmenetelmiin sekä luokittelujen keskinäisen vertailun menetelmiin. Lisäksi erilaisuusmittojen määrittäminen data-olioiden välillä on tärkeä teema. Diplomityön toisessa osassa menetelmiä ja määritelmiä sovelletaan Euroopan nisäkkäiden esiintymistä käsittelevään tietokantaan. Käytännön tutkimusongelmana on selvittää tukeeko nisäkkäiden esiintyminen hierarkkisen luokittelun mallia.

Diplomityössä analysoidaan levinneisyyteen perustuvien etäisyysmittojen käyttäytymistä nisäkkäiden välillä. Tämän pohjalta muodostetaan joukko hierarkkisia luokitteluja käyttäen sekä kokoavaa klusterointia että ahnetta hierarkiapuun hakustrategiaa. Hierarkkisen luokittelumallin sopivuutta nisäkäsaineistoon arvioidaan käyttäen Monte Carlo- ja Bootstrap -menetelmiä, joista molemmat perustuvat alkuperäisen aineiston uudelleenotantaan. Tulokset antavat uskottavan, maantieteellisen jakoon perustuvan hierarkkisen luokittelun aineiston nisäkkäille.

Avainsanat: hierarkia, taksonomia, ultrametrinen, binääripuu, etäisyysmitta, klusterointi, puiden vertailu, mallin hyvyys

Acknowledgments

During the process of writing this Master's thesis I have received help and support from several people. For this I am very grateful. Especially, I would like to thank my instructor Professor Heikki Mannila. In addition, I would like to thank all the other people of the Pattern Discovery group and of course Professor Mikael Fortelius. The working environment has been very inspiring and educative. Finally, my thoughts go to my parents, my grandparents, my sister and my girlfriend Katri for their love and support.

Otaniemi, September 30, 2005

Hannes Heikinheimo

Contents

Chapter 1

Introduction

1.1 Background

Complex structures in nature and in society are frequently modeled and managed with hierarchies. For engineers and scientists hierarchies are a tool used for abstraction and classification. For example, a software engineer uses hierarchical abstraction to build and manage complex computer programs. A biologist uses hierarchical classification to understand the diverse relationships between organisms.

Taxonomic hierarchies are hierarchies that define a set of subclass relations between categories of objects [44, p. 323]. In other words, a taxonomic hierarchy is a hierarchical classification of objects from the abstract to the specific. The word taxonomy originates from the Greek word *taxinomia* meaning order distribution or order law [45]. For instance, a biological classification of organisms into Species, Genus, Family and so forth forms a taxonomic hierarchy. A taxonomy itself, however, need not be a hierarchy, but can be organized in a network-like structures as well.

In this thesis we are concerned with the inference of taxonomic hierarchies. The motivation for the work comes from a practical starting point: we have been introduced an interesting data set consisting of occurrence ranges of European land mammals. The data gives us an opportunity to apply and review techniques related to the topic of taxonomic hierarchy inference. The work can be considered to fall in the discipline of *data mining*.

In the following of this introductory chapter we set the premises of our work. In the next section we give a more mathematical working definition for a taxonomic hierarchy, followed by a brief general description of data mining in Section 1.3. In Section 1.4 we formulate the problem statement of this work. Section 1.5 gives a summary on how the thesis will continue in the

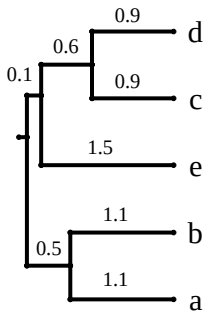


Figure 1.1: A taxonomic hierarchy tree for objects a, b, c, d and e. Horizontal lines represent branches (subclass relations) and vertical lines represent nodes (categories). The lengths of the branches are marked with decimal numbers.

following chapters.

1.2 Definition for a taxonomic hierarchy

Mathematically, a taxonomic hierarchy can be represented with a rooted tree structure. The set of subclass relations in a hierarchy can be associated with the branches of the tree and categories with the nodes of the tree. In such a taxonomy tree the root node depicts the most abstract category under which all objects in the system belong to. Nodes branching onwards form the root define more specific subcategories in which only a subset of the objects belong to. The leafs of the tree define the most specific categories identifying each object exactly.

Figure 1.1 shows an example of a taxonomic hierarchy tree constructed for objects a, b, c, d and e. The tree is composed of the branches (subclass relations) represented with horizontal lines and nodes (categories) represented with vertical lines. The tree is said to be *labeled* with the objects. Note also, that each internal node of the trees has strictly two branches. Trees like this are called *binary trees*. When representing a taxonomic hierarchy with a binary tree a category always has two immediate subcategories. For simplicity we use binary trees in the study of taxonomic hierarchies.

To define a scale quantifying a *level of hierarchy* on a taxonomic hierarchy, branch lengths can be assigned to the tree representation. A branch length between a category (node) and subcategory (subnode) gives us the information of how far from each other the two categories are in terms of hierarchy. The absolute level of hierarchy for a category is then the sum of branch lengths from the root of the tree to the node of the category. Respec-

tively, taking a line and cross cutting all branches at a certain distance from the root, defines a specific level of hierarchy on a tree. The nodes immediately under the cross cutting line define the categories of the corresponding level of hierarchy. Trees with specified branch lengths are called *weighted trees*. The tree of Figure 1.1 is weighted: the branches of the tree have non-equal lengths.

Fundamentally, a taxonomic hierarchy on a set of objects is a model on how the similarities between the objects are organized. Given a certain level of abstraction, objects in a same subcategory should always be more similar to each other than objects in a different subcategory. Also, grouping a set of objects to a category forces the objects inside a category to behave collectively similarly in respect of objects outside the category. In addition this must hold true at each level of the hierarchy.

In order to formulate this mathematically, we quantify the similarity between two objects $\mathbf{x}$ and $\mathbf{y}$ with a function $d(\mathbf{x}, \mathbf{y}) \geq 0$, so that the smaller the value of $d(\mathbf{x}, \mathbf{y})$ gets the more similar $\mathbf{x}$ and $\mathbf{y}$ are. In a tree, $d(\mathbf{x}, \mathbf{y})$ is the sum of lengths of the branches that form the unique path from $\mathbf{x}$ to $\mathbf{y}$, via the node defining the most low hierarchy category common to both $\mathbf{x}$ and $\mathbf{y}$.

Definition 1.1 A taxonomic hierarchy for a set of objects D is defined by a rooted binary tree $\mathcal{T}$, labeled with the objects of D and having branch lengths such that the distance between any objects $\mathbf{x}$, $\mathbf{y}$, $\mathbf{z} \in D$ in the tree satisfies the constraint

$$d(\mathbf{x}, \mathbf{y}) \leq \max\left(d(\mathbf{x}, \mathbf{z}), d(\mathbf{y}, \mathbf{z})\right) \tag{1.1}$$

called the *ultrametric* property.

To elucidate what ultrametricity means in terms of $\mathcal{T}$, consider a subcategory C and three objects $\mathbf{x}$, $\mathbf{y}$ and $\mathbf{z}$, so that only $\mathbf{x}$ and $\mathbf{y}$ belong to the category C. The distances of the tree objects in $\mathcal{T}$ satisfy

$$d(\mathbf{x}, \mathbf{y}) \leq d(\mathbf{x}, \mathbf{z}) = d(\mathbf{y}, \mathbf{z}), \tag{1.2}$$

also known as the *three point condition*. In other words, the similarity between the same category objects $\mathbf{x}$ and $\mathbf{y}$ is at least as great as the similarity of $\mathbf{x}$ and $\mathbf{y}$ between the object $\mathbf{z}$ in some other category. In addition, objects $\mathbf{x}$ and $\mathbf{y}$ inside C are regarded equally similar to $\mathbf{z}$. From ultrametricity it also follows that the paths from the root node to all the leaf nodes in $\mathcal{T}$ will have the same length. Trying Definition 1.1 on the tree of Figure 1.1 will help perceive its meaning. [23, p 449-458]

It is important to notice that the ultrametric property is more restrictive than what an arbitrary weighted, labeled tree structure would imply. A tree having branch lengths that do not satisfy the ultrametric property is called an *additive tree*. Additivity, however, does not guarantee that objects in a same category are always more similar between each other and less similar to objects in a different category.

1.3 Data mining

This thesis has been done in the spirit of *knowledge discovery*, often referred as *data mining*. Data mining is an interdisciplinary research area combining techniques among others from artificial intelligence, pattern recognition, statistics and databases. It develops general purpose methods applicable for finding interesting and useful knowledge from large real life collections of data. [25]

Data mining techniques are often applied to a data set consisting of some measurements taken from a given application domain. We can think of the measurements as a collection of *observations* over a set of *attributes*. The set of n attributes for m observations gives us an $m \times n$ matrix called the *data matrix*. For example, if the domain is market analysis, one observation can be the contents of a shopping cart of some customer at a supermarket cash register. The attributes can be the products on sale in the supermarket. The overall data would then be the a set of products purchased by a set of customers in a supermarket during some period of time.

In the previous section we defined a taxonomic hierarchy for a set of objects. From now on, with objects we shall mean either observations (rows) or attributes (columns) in a data matrix depending on the context. For a data matrix having only numerical measurements, both, observations and attributes, can be seen as vectors in the row or column space of the data matrix.

1.4 Research problem

We have been introduced an interesting data set of occurrence ranges of European land mammals. Our experimental interests are in finding out what kind of spatial structure the data has. Especially we are interested in how the mammal species co-occur and whether some hierarchical structure is present. Also, we would like to identify in what the possible structure it is related to.

Problem statement 1) What kind of taxonomic hierarchies does the occurrence of species in the data imply and 2) is a hierarchical model justified?

The problem statement divides our task into two parts: an inference step and a justification step for taxonomic hierarchies. For the inference step we have to consider how similarities can be defined between objects (species) in the data and how inference of taxonomic hierarchies can be done based on these similarities. For the justification step, we must assess how well the similarities fit in fact a hierarchical taxonomy structure.

The field of hierarchical taxonomy inference is extremely vast. In our study we will concentrate of techniques sometimes referred as distance-based methods. In addition, because the mammal occurrence data is two valued (presence/absence), our interests lie on 0-1 data. Concerning hierarchies, our focus is on the ultrametric model according to Definition 1.1.

1.5 The structure of this thesis

This thesis is divided into two parts. The first part is a domain independent review on methods available for inference and assessment of taxonomic hierarchies. It consists of Chapters 2, 3 and 4. The second part, discussed in Chapter 5, covers the experimental contribution of this work addressed to answer our research problem. We summarize the structure of this thesis as follows:

In Chapter 2 we are concerned with the notion of similarity in the form of distance between two objects. We discuss some general properties that a distance measure should satisfy and review some basic definitions. The discussion will mostly concentrate on objects represented by binary vectors.

In Chapter 3 we cover two basic methods of taxonomic hierarchy inference: agglomerative clustering and the least squares method. Both methods base their inference on a distance function defined between the objects for which a hierarchy is wished to be constructed.

In Chapter 4 we will take a look at how distance between different taxonomic hierarchies can be measured and how the validity of a hierarchical ultrametric model can be assessed for a given data.

Finally, in Chapter 5 we give a more precise description of the applied mammal data set and present the experimental results of our work. We conclude our work in the end of Chapter 5.

Chapter 2

Measuring distance

A categorization or a grouping of objects can not be done without a measure of similarity of some sort. To establish the grounds based on which objects are grouped, we need a logic that deems some objects similar and some objects dissimilar. The logic depends on the aspects we consider important or want to highlight in the categorization.

For a set of objects represented by numerical vectors in the space of observations and attributes, similarity is defined by the complementary notion of distance. When inferring taxonomic hierarchies from such numerical objects, the aspects considered important in the categorization are encoded to the definition of distance. Because these aspects differ from application to application, there exists a large variety of definitions to choose from.

In this chapter we are concerned about distance inside two very general classes of definitions: internal and external [7]. Furthermore, our discussion will mostly concentrate on objects represented by binary vectors. We will start, however, by summarizing some general properties that a distance definition should satisfy. These are called metric properties of a distance.

2.1 General properties

Reflecting on our everyday intuition, we expect a distance from a place X to a place Y to have some basic properties. For instance, if the distance between X and Y is zero, X and Y should to be the same place. Furthermore, a negative distance does not seem reasonable. Also, the distance measure form X to Y via some place Z is expected to have a greater value in comparison to the bee line distance measure from X to Y. A distance function satisfying these expectations is called a *metric*.

Definition 2.1 A distance function d is a *metric* if for all objects $\mathbf{x}$, $\mathbf{y}$ and $\mathbf{z}$ the following conditions hold:

1. $d(\mathbf{x}, \mathbf{y}) \geq 0$;
2. $d(\mathbf{x}, \mathbf{y}) = 0$ if and only if $\mathbf{x} = \mathbf{y}$;
3. $d(\mathbf{x}, \mathbf{y}) = d(\mathbf{y}, \mathbf{x})$;
4. $d(\mathbf{x}, \mathbf{y}) \leq d(\mathbf{x}, \mathbf{z}) + d(\mathbf{z}, \mathbf{y})$.

Definition 2.1 generalizes the spatial properties of the *Euclidean space*. In general, the properties of a metric are desirable because of their intuitiveness and practicality; for a metric distance function we always know that if $\mathbf{x}$ is close to $\mathbf{z}$ and $\mathbf{z}$ is close to $\mathbf{y}$, $\mathbf{x}$ has to be close to $\mathbf{y}$. This information can be crucial in the effective solving of a number of computational problems related, for instance, to searching or data retrieval.

However, in some cases the use of distance measures with reduced metric properties can be fruitful as well. If we accept a relaxation of certain conditions, we might attain some additional and otherwise unexpected knowledge about the relationships of the objects we are investigating.

Definition 2.2 A distance function d is a *pseudometric* if for all objects $\mathbf{x}$, $\mathbf{y}$ and $\mathbf{z}$ the following conditions hold:

1. $d(\mathbf{x}, \mathbf{y}) \geq 0$;
2. $d(\mathbf{x}, \mathbf{x}) = 0$;
3. $d(\mathbf{x}, \mathbf{y}) = d(\mathbf{y}, \mathbf{x})$;
4. $d(\mathbf{x}, \mathbf{y}) \leq d(\mathbf{x}, \mathbf{z}) + d(\mathbf{z}, \mathbf{y})$.

Definitions 2.1 and 2.2 differ only in condition 2. A pseudometric should assign a distance of zero to two identical objects, but unlike a metric it may also give a distance of zero to two non-identical objects. Of course, from the applications point of view, a zero distance should be given only to objects having identical features in respect to what we are measuring, but such objects are not required to be strictly the same.

Definition 2.3 A distance function d is a *semimetric* if for all objects $\mathbf{x}$, $\mathbf{y}$ and $\mathbf{z}$ the following conditions hold:

1. $d(\mathbf{x}, \mathbf{y}) \geq 0$;

2. $d(\mathbf{x}, \mathbf{y}) = 0$ if and only if $\mathbf{x} = \mathbf{y}$;

3. $d(\mathbf{x}, \mathbf{y}) = d(\mathbf{y}, \mathbf{x})$.

A semimetric function d satisfies the three first conditions of a metric distance function but not necessarily the triangle inequality of condition 4. From the three above mentioned metric definitions a semimetric is the most relaxed, but also the most difficult to picture spatially. In cases were the fulfillment of the triangle inequality is less important, a semimetric distance function with other good application related qualities can yield better results than the more restricted metric functions.

2.2 Internal measures of distance

Internal distance is a measure of dissimilarity based on pairwise comparison of the vector components of $\mathbf{x}$ and $\mathbf{y}$. This is perhaps the most intuitive and surely the most common way to measure distance between $\mathbf{x}$ and $\mathbf{y}$. In this section we will go through some basic definitions.

2.2.1 Binary coefficients

Let objects $\mathbf{x} = (x_1, ..., x_n)^T$ and $\mathbf{y} = (y_1, ..., y_n)^T$ be two vectors, so that for $k = 1, ..., n$ the vector components x_k and y_k take values in $\{0, 1\}$. In other words, $\mathbf{x}$ and $\mathbf{y}$ are two n dimension binary vectors. For convenience in defining distance measures between $\mathbf{x}$ and $\mathbf{y}$ we use a 2×2 *contingency matrix*

$$M(\mathbf{x}, \mathbf{y}) = \begin{bmatrix} m_{00} & m_{01} \\ m_{10} & m_{11} \end{bmatrix}, \tag{2.1}$$

where for $i, j \in \{0, 1\}$ the quantity m_{ij} is the number of components k such that $x_k = i$ and $y_k = j$.

Example Let $\mathbf{x} = [0, 0, 1, 1]^T$ and $\mathbf{y} = [1, 0, 1, 1]^T$ then

$$M(\mathbf{x}, \mathbf{y}) = \begin{bmatrix} 1 & 1 \\ 0 & 2 \end{bmatrix}.$$

Perhaps the simplest binary distance measure having metric properties is the *Hamming distance*

$$d_h(\mathbf{x}, \mathbf{y}) = \frac{m_{01} + m_{10}}{m_{00} + m_{01} + m_{10} + m_{11}}, \tag{2.2}$$

defined as the fraction of places where the vector components of the objects differ [24]. The measure assumes that both m_{00} and m_{11} contribute to the similarity of the two vectors. In other words large values of both m_{00} and m_{11} decrease the value of the distance.

However, the question of whether a large m_{00} is really related to the similarity of two objects varies from domain to domain. For instance, say the values 1 and 0 are indicators of presence and absence of certain two very rare objects. Hamming distance will automatically assign a small distance value based on a large m_{00}, although rareness itself would not necessarily be a justified reason for similarity. When common 0 scores are irrelevant a good alternative is the *Jaccard distance*

$$d_j(\mathbf{x}, \mathbf{y}) = 1 - \frac{m_{11}}{m_{01} + m_{10} + m_{11}} = \frac{m_{10} + m_{01}}{m_{01} + m_{10} + m_{11}}, \tag{2.3}$$

where $m_{11}/(m_{01}+m_{10}+m_{11})$ is the so called *Jaccard coefficient* [29]. Function d_J omits m_{00} altogether but like d_H it has the metric properties of Definition 2.1 [22].

When removing m_{00} from the equation 2.2, more weight in proportion is put on m_{01} and m_{10}. To readjust the weighting of difference to the level of Hamming distance, we can use the *Dice distance* [9]

$$d_d(\mathbf{x}, \mathbf{y}) = 1 - \frac{2m_{11}}{m_{01} + m_{10} + 2m_{11}} = \frac{m_{01} + m_{10}}{m_{01} + m_{10} + 2m_{11}}, \tag{2.4}$$

which weights the values of m_{01} and m_{10} by half of what the Jaccard distance does.

In some applications even more emphasis on common scores of 1 may be needed. For instance, when the number of 1 scores is expected to be very different in the compared vectors $\mathbf{x}$ and $\mathbf{y}$, either m_{01} or m_{10} will dominate the distance equation of d_j and d_d. In addition, if the dimension n is large, the effect grows giving automatically large values to the measures. To reduce the possible domination of m_{10} or m_{01} one option is to use the *Second Kulczynski distance* [6]

$$d_k(\mathbf{x}, \mathbf{y}) = 1 - \frac{1}{2}\left(\frac{m_{11}}{m_{01} + m_{11}} + \frac{m_{11}}{m_{10} + m_{11}}\right). \tag{2.5}$$

The idea of the measure is to average the effect that the value of m_{10} and m_{01} have in proportion to m_{11}. Hence, if the proportion of m_{11} is large in comparison of either m_{10} or m_{01}, the Second Kulczynski distance will give more weight to common scores of 1 that what the Jaccard or Dice distance would give. However, in the extreme case of, say, $m_{01} \ll m_{11} \ll m_{10}$, d_k starts to approach values close to 0.5. The Second Kulczynski distance is a semimetric [22].

2.2.2 Correlation and Cosine distance

Linear association between the components of two real valued objects $\mathbf{x}$ and $\mathbf{y}$ can be measured with *covariance*. Let the mean value of the vector components $x_1, .., x_n$ of object $\mathbf{x}$ be

$$\bar{x} = E(x) = \frac{1}{n}\sum_{k=1}^{n} x_k. \tag{2.6}$$

For objects $\mathbf{x}$ and $\mathbf{y}$, their components covary if x_k and y_k are expected to have the same tendency of being above or below their means $\bar{x}$ and $\bar{y}$. We write

$$cov(\mathbf{x}, \mathbf{y}) = \sigma_{\mathbf{xy}} = \frac{1}{n}\sum_{k=1}^{n}(x_k - \bar{x})(y_k - \bar{y}) \tag{2.7}$$

for the covariance of the objects $\mathbf{x}$ and $\mathbf{y}$. In analogy, the *variance* and *standard deviation* of object $\mathbf{x}$ are $\sigma_{\mathbf{xx}}$ and $\sqrt{\sigma_{\mathbf{xx}}}$ respectively and denoted hence by $\sigma^2_{\mathbf{x}}$ and $\sigma_{\mathbf{x}}$.

It is important to note, that the measure of covariance is scale dependent. In other words, if the values of, say $\mathbf{x}$, were to be multiplied by a constant, the value of $cov(\mathbf{x}, \mathbf{y})$ would be affected. For scale invariance we can standardize the coefficient between 1 and -1 by dividing it by the deviation of $\mathbf{x}$ and $\mathbf{y}$. We call the standardized covariance

$$corr(\mathbf{x}, \mathbf{y}) = \frac{\sigma_{\mathbf{xy}}}{\sigma_{\mathbf{x}}\sigma_{\mathbf{y}}} = \frac{\sum_{k=1}^{n}(x_k - \bar{x})(y_k - \bar{y})}{\sqrt{\sum_{k=1}^{n}(x_k - \bar{x})^2}\sqrt{\sum_{k=1}^{n}(y_k - \bar{y})^2}}, \tag{2.8}$$

the *correlation* of objects $\mathbf{x}$ and $\mathbf{y}$. If $corr(\mathbf{x}, \mathbf{y})$ is close to zero, $\mathbf{x}$ and $\mathbf{y}$ are said to be *uncorrelated*. [43, p. 45-51]

Correlation can be used to measure association between the components of two binary objects as well. If $\mathbf{x}$ and $\mathbf{y}$ are binary valued, function 2.8 is equivalent to the form

$$corr(\mathbf{x}, \mathbf{y}) = \frac{m_{11}m_{00} - m_{01}m_{10}}{\sqrt{(m_{11} + m_{10})(m_{00} + m_{10})}\sqrt{(m_{11} + m_{01})(m_{00} + m_{01})}}. \tag{2.9}$$

Expression 2.9 is referred sometimes as the *Phi coefficient* of $\mathbf{x}$ and $\mathbf{y}$ [6, 49].

Because correlation is a measure of association of variance between the components of $\mathbf{x}$ and $\mathbf{y}$, it can be considered to be related to the proximity of $\mathbf{x}$ and $\mathbf{y}$ as well. Consequently, by subtracting the correlation coefficient from 1, we get a distance measure

$$d_{corr}(\mathbf{x}, \mathbf{y}) = 1 - corr(\mathbf{x}, \mathbf{y}), \tag{2.10}$$

that in the binary case is a semimetric [22].

Correlation of $\mathbf{x}$ and $\mathbf{y}$ is also related the cosine function

$$\cos\theta_{\mathbf{xy}} = \frac{\mathbf{x}^T\mathbf{y}}{\sqrt{(\mathbf{x}^T\mathbf{x})(\mathbf{y}^\mathbf{T}\mathbf{y})}}, \tag{2.11}$$

where $\theta_{\mathbf{xy}}$ is the angle between the object vectors $\mathbf{x}$ and $\mathbf{y}$. By replacing $\mathbf{x}$ and $\mathbf{y}$ in 2.11 with $\mathbf{x}-\bar{x}\mathbf{i}$ and $\mathbf{y}-\bar{y}\mathbf{i}$, where $\mathbf{i}$ is a vector of 1's, we get a form

$$\cos\theta_{(\mathbf{x}-\bar{x}\mathbf{i})(\mathbf{y}-\bar{y}\mathbf{i})} = \frac{(\mathbf{x}-\bar{x}\mathbf{i})^T(\mathbf{y}-\bar{y}\mathbf{i})}{\sqrt{(\mathbf{x}-\bar{x}\mathbf{i})^T(\mathbf{x}-\bar{x}\mathbf{i})}\sqrt{(\mathbf{y}-\bar{y}\mathbf{i})^T(\mathbf{y}-\bar{y}\mathbf{i})}} \tag{2.12}$$

equivalent to 2.8. [43, p. 45-51]

The connection between cosine and correlation means, for one thing, that if $\bar{x}$ and $\bar{y}$ are zero, expressions 2.8 and 2.11 produce the same value. Of course, for a binary object $\mathbf{x}$, the mean $\bar{x}$ can be zero only if $\mathbf{x}$ has a 0 score in each component. The major difference from the binary point of view is, however, that cosine ignores common 0 scores whereas correlation does not. This is easy to see by expressing function 2.11 with the entries of the 2×2 contingency matrix. We get

$$\cos\theta_{\mathbf{xy}} = \frac{m_{11}}{\sqrt{(m_{11}+m_{10})}\sqrt{(m_{11}+m_{01})}}, \tag{2.13}$$

a more stripped-down version of the Phi coefficient. To utilize cosine as a distance measure we define the *cosine distance*, as

$$d_{cos} = 1 - \cos\theta_{\mathbf{xy}}. \tag{2.14}$$

For binary objects, d_{cos} is also a semimetric [22].

2.3 External measures of distance

External distances are defined based on how similarly $\mathbf{x}$ and $\mathbf{y}$ are placed in respect to other objects in the data set. In other words, $\mathbf{x}$ and $\mathbf{y}$ are not compared directly with each other, but through the similarity of distance to a set of reference objects. We call the set of reference objects, the *probe set*, and denote it with P.

As an example, consider the setting of finding similarities between the products customers buy in super markets. For instance, one would expect different kinds of soft drink products, say Coke and Pepsi to have a small distance values. However, customers seldom buy Coke and Pepsi together. This

results to a large distance when taking into account pairwise co-occurrence. By contrast the thing that makes these products similar are the other products bought with them, chips for instance. External measures try to make use of this fact and take into account the occurrence of other items in the data set. [7]

2.3.1 Probe set comparison

Let R be a set of attributes and let objects $\mathbf{x}, \mathbf{y} \in R$. Furthermore, let $P \subset R$ be a set of probe attributes, so that $\mathbf{x}, \mathbf{y} \notin P$. Now, consider an $m \times |R|$ binary data matrix D where the columns represent the attributes R. We define the submatrix $D_P^{\mathbf{x}}$ of D by the rows of D where object $\mathbf{x}$ has an entry 1 and by the respective columns that represent the attributes in P. Furthermore, we define $f_{\mathbf{x}}$ as a multivariate distribution on $\{0,1\}^{|P|}$. With the notation $f_{\mathbf{x}}(\rho)$ we mean the relative frequency of $\rho \in \{0,1\}^{|P|}$ in the rows of matrix $D_P^{\mathbf{x}}$. Consider also $D_P^{\mathbf{y}}$ and $f_{\mathbf{y}}$ for object $\mathbf{y}$ respectively.

Now, the idea is to look at the distribution of probe attribute occurrence on the rows of $D_P^{\mathbf{x}}$ and compare it to the respective distribution of $D_P^{\mathbf{y}}$. If the distributions are similar, $\mathbf{x}$ and $\mathbf{y}$ are close to each other in the sense of external distance. To apply this idea, we can use, for instance the *Kullbach-Leibler distance*, which is a measure often used for comparing two frequency distributions [33]. For distance between $\mathbf{x}$ and $\mathbf{y}$ we get

$$d_{kl}(\mathbf{x}, \mathbf{y}) = -\sum_{\rho} f_{\mathbf{x}}(\rho) \log \frac{f_{\mathbf{x}}(\rho)}{f_{\mathbf{y}}(\rho)}. \tag{2.15}$$

The Kullbach-Leibler distance is known to yield only nonnegative values [47, p. 647]. However, the condition 2 of Definition 2.1 is violated. Also, it is not symmetric, although a symmetric measure can be obtained with the form $d_{kl}(\mathbf{x}, \mathbf{y}) + d_{kl}(\mathbf{y}, \mathbf{x})$.

The downside of the Kullbach-Leibler distance is that in order to compute it one has to iterate through the $2^{|P|}$ size set of different $\rho \in \{0,1\}^{|P|}$. With large sets of P this becomes a problem. A way to avoid the computational complexity is to compare $\mathbf{x}$ and $\mathbf{y}$ respect to only one probe object of P at a time [7]. In comparison to the Kullbach-Leibler distance, this will of course lose information concerning interaction of probes, but will be easier to compute. We define the *Probe distance* as

$$d_p(\mathbf{x}, \mathbf{y}) = \sum_{\mathbf{z} \in P} |d_I(\mathbf{x}, \mathbf{z}) - d_I(\mathbf{y}, \mathbf{z})|, \tag{2.16}$$

where d_I is some internal distance measure.

From the sum term in function 2.16 we can see that the distance will build up if objects $\mathbf{x}$ and $\mathbf{y}$ have very different distances respect the probe objects. Other properties of d_p will naturally depend on the choice of d_I. This makes probe distance a versatile distance measure; d_I can be chosen according to the domain of application. At least, if d_I is equal to the Jaccard distance, d_p has the properties of a pseudometric.

2.3.2 Selecting probes

It is clear that different probe sets produce different distance measures. This is why the selection of probes should somehow reflect the aspects of data that we thing are interesting and essential. As an example related to the super market setting discussed earlier, we could be especially interested in how similarly different soft drink products are bought with different potato chips. In this case it would be natural to select a probe set consist of only potato chip products.

Another approach in selecting a probe set is to try to choose the probes so that the resulting distance function will fulfill some à prior property we know or want the objects to have. This could be, say, ordinal information of the distances of some subset of objects. By finding such a probe set we could then infer what the à prior conditions mean regarding the unknown distances. In any case, in these issues the knowledge of the domain expert is useful. [40, p. 32-33]

Chapter 3

Generating taxonomic hierarchies

Taxonomic hierarchies are widely used to model data both in scientific and business related domains. Examples of applications can be found among others from systematic biology, medicine, psychology, market research and artificial intelligence. This means that the amount of literature on methods for generating taxonomic hierarchies is huge and the number of different schools and paradigms are various.

As a scratch on the surface, we will discuss two basic methods of taxonomic hierarchy inference: agglomerative clustering and the least squares method. Both methods base their inference on a quantitative distance function d defined between the objects in the hierarchy. Such methods are referred, especially in the context of phylogenetic inference, as *distance-based methods* [18, 32].

3.1 Agglomerative clustering

3.1.1 Cluster analysis in general

Cluster analysis refers to methods that try to uncover groups or *clusters* in data. Especially, most applications of cluster analysis often seek to find *crisp* groupings. With this we mean descriptive *partitionings* for the data objects so that each object belongs to a single group, and the complete set of groups contains all objects. We define a *clustering* as follows [47, p. 402]:

Definition 3.1 Let $D = \{\mathbf{x}_1, .., \mathbf{x}_n\}$ be the set of data objects forming our data matrix. A clustering of D denoted by $\mathfrak{S}$ is a partitioning of D into m sets or *clusters* $C_1, ..., C_m$ by some similarity criterion, so that the following conditions hold:

1. $C_i \neq \emptyset, i = 1, ..., m$;
2. $\cup_{i=1}^{m} C_i = D$;
3. $C_i \cap C_j = \emptyset$, $i, j = 1, ..., m$ and $i \neq j$.

The essential point of how a clustering differs from a normal partitioning is that the objects contained in the same cluster should be more similar to each other and less similar to the objects in the other clusters. However, the definition does not specify the similarity criterion explicitly. This is because it may differ dependentlyx on the aspects of data we are interested in, e.g. some specific attributes of data. Also, the properties we expect our data to have effect our choice of the criterion. Hence, defining it more accurately comes back to the discussion of chapter 2 of defining appropriate distance measures for the examined objects.

In addition to distance, the similarity criterion is related to the expected *shape* of the clusters. Considering the data objects as points in an m dimensional space $\mathbb{R}^m$, with shape we mean the spatial structure of the clusters in $\mathbb{R}^m$. Figure 3.1 illustrates this point. The clustering of the same set of data points can have very different results when the shape of clusters is considered differently. Again, the domain of the clustering application marks the choosing of the appropriate cluster shape.

The third remark about Definition 3.1 is related to the number of different possible clusterings. For n data objects the number of different partitionings is equal to

$$B(n) = \sum_{i=1}^{n} S(n, i), \tag{3.1}$$

where S is defined as the *Stirling number* (of the second kind) [3, p. 126]:

$$S(n, m) = \begin{cases} mS(n-1, m) + S(n-1, m-1), & \text{if } 2 \leq m \leq n-1; \\ 1, & \text{for } m = 1, n. \end{cases} \tag{3.2}$$

The Stirling number is known to grow very rapidly already for moderate n and m. For instance,

$$S(20, 4) = 45232115901 \text{ and } S(25, 8) = 690223721118368580.$$

In other words, the option of going through all possible partitions to find the best clustering is not a realistic one; more intelligent strategies are needed. [47, p. 429-430]

The study of different clustering criteria and alternative clustering strategies give birth to a myriad of different algorithms developed in the past

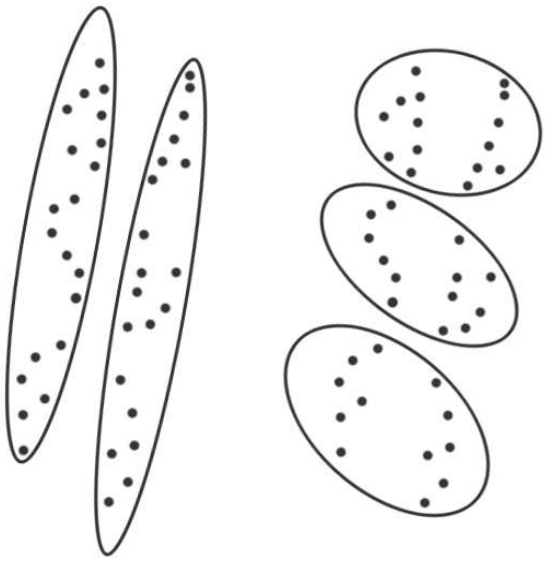


Figure 3.1: Two different clustering results for the same set of points in a two dimensional space; the criterion used in clustering (a) assumes *elongated* clusters as opposed to the more *spherical* or *compact* ones in clustering (b).

decades. To work around the complexity of partitioning the basic idea for all of these methods is to try to examine only a fraction of all possible clusterings of the data while optimizing some similarity criterion. An extensive overview of different clustering schemes can be found among others from [13], [31] and [47].

3.1.2 The agglomerative algorithm

Consider a taxonomic hierarchy that categorizes biological organism into species, genus, family and so fort. From the clustering point of view the different levels of the taxonomy form a clustering in the sense of Definition 3.1: every organism is grouped precisely into one category at each level. A taxonomic hierarchy can thus be regarded as a hierarchy of multiple clusterings, where every cluster on a lower level is a subsets of some cluster on a higher level. We say that a clustering $\mathfrak{S}_i$, which contains k clusters, is *nested* in the clustering $\mathfrak{S}_j$ containing $r < k$ clusters, if each cluster in $\mathfrak{S}_i$ is a subset of a set in $\mathfrak{S}_j$. In this case we denote $\mathfrak{S}_i \sqsubset \mathfrak{S}_j$.

Example For a data set $D = \{\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3, \mathbf{x}_4\}$ and two clusterings $\mathfrak{S}_1 = \{\{\mathbf{x}_1, \mathbf{x}_3\}\{\mathbf{x}_2\}\{\mathbf{x}_4\}\}$ and $\mathfrak{S}_2 = \{\{\mathbf{x}_1, \mathbf{x}_3\}\{\mathbf{x}_2, \mathbf{x}_4\}\}$, we denote $\mathfrak{S}_1 \sqsubset \mathfrak{S}_2$.

Algorithm 3.1

Input: A data set $D = \{\mathbf{x}_1, ..., \mathbf{x}_n\}$.
Output: A hierarchy $\mathfrak{S}_0 \sqsubset ... \sqsubset \mathfrak{S}_{n-1}$ of nested clusterings.

```
1: i = 0;
2: 𝔖_0 = {C_1 = {x_1}, ..., C_n = {x_n}};
3: while |𝔖_i| > 1 do
4:     (C_g, C_h) = argmin_{C_j,C_k} (𝒟(C_j, C_k) | C_j, C_k ∈ 𝔖_i, j ≠ k);
5:     𝔖_{i+1} = 𝔖_i \ {C_g} \ {C_h} ∪ {C_g ∪ C_h};
6:     i=i+1;
7: end;
```

Definition 3.2 Let $D = \{\mathbf{x}_1, ..., \mathbf{x}_n\}$ be a data set of n objects and let $\{\mathfrak{S}_0, ..., \mathfrak{S}_{n-1}\}$ be a set of n clusterings of D, so that for $i = 0, ..., n-1$ each clustering $\mathfrak{S}_i$ consists of $n - i$ clusters and $\mathfrak{S}_i \sqsubset \mathfrak{S}_{i+1}$, for $i < n - 1$. We call $\mathfrak{S}_0 \sqsubset ... \sqsubset \mathfrak{S}_{n-1}$ a *hierarchy of nested clusterings.*

Methods that initialize from $\mathfrak{S}_0$ and use merger of clusters to produce a hierarchy of nested clusterings are called *agglomerative* algorithms. The general idea is to produce the new clustering of the next level of hierarchy by merging the two most similar clusters of the current level according to some link criterion $\mathcal{D}$ defining distance between two clusters. The algorithm 3.1 defines the agglomerative procedure [25, p. 311].

Algorithm 3.1 starts, on rows 1-2, by assigning the first clustering $\mathfrak{S}_0$ to consist of one data object per one cluster. On row 3 the while loop iterates until the last clustering, containing only a single cluster, has been produced. Inside the loop the statement of row 4 assigns C_g and C_h as the two clusters having minimal distance $\mathcal{D}$ in $\mathfrak{S}_i$. On row 5 the next level clustering $\mathfrak{S}_{i+i}$ is generated, by first removing C_g and C_h from the previous level clustering $\mathfrak{S}_i$ and than adding the merger of the two clusters to it. The while loop is repeated $n - 1$ times and for each iteration $\binom{n-i}{2}$ cluster pairs are examined. The total amount of operations is thus:

$$\begin{aligned}\sum_{i=0}^{n-2}\binom{n-i}{2} &= \sum_{i=2}^{n}\binom{i}{2} = \sum_{i=2}^{n}\frac{i!}{2!(i-2)!} = \frac{1}{2}\sum_{i=2}^{n}(i^2 - i) \\ &= \frac{1}{2}(\sum_{i=2}^{n} i^2 - \sum_{i=2}^{n} i) = \frac{n^3 - n}{6}.\end{aligned}$$

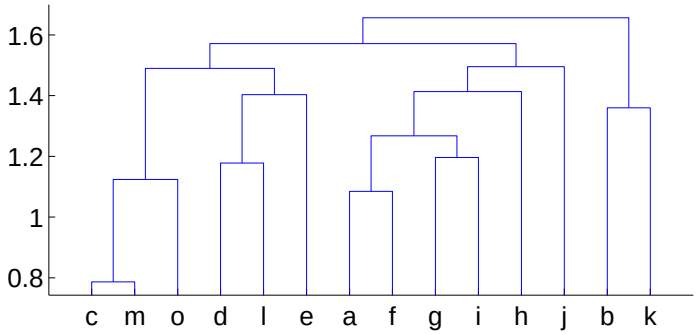


Figure 3.2: A dendrogram tree representing the outcome of an agglomerative cluster algorithm for objects a,..,o. The branch lengths of the tree are defined by the scale of the y-axis. Respectively, the nodes of the dendrogram are the levels of hierarchy were two clusters merge.

The time complexity of the algorithm is therefore $O(n^3)$. Of course, the precise complexity of the algorithm is dependent on how the statement of row 4 is implemented [47, p. 451].

A tree-like illustration of the taxonomic hierarchy outputted by an agglomerative algorithm is called a *dendrogram*. In fact, a dendrogram is a representation of a weighted, rooted, labeled, binary tree. Figure 3.2 shows an example of a dendrogram outputted for objects a,...,o. On the figure, the y-axis depicts the level of hierarchy in the taxonomy. Thus, the branch lengths are defined by the scale of the y-axis. Respectively, the nodes of the dendrogram are the levels of hierarchy were two clusters (categories) merge.

The information of a dendrogram tree can be encoded as an $n \times n$ distance matrix as well.

Definition 3.3 Let $D = \{\mathbf{x}_1, .., \mathbf{x}_n\}$ be a set of data objects and $C_g, C_h \subset D$ two clusters that merge at some point of the execution of algorithm 3.1. For all objects $\mathbf{x}_i \in C_g$ and $\mathbf{x}_j \in C_h$, we assign $d_{coph}(\mathbf{x}_i, \mathbf{x}_j)$ as the distance $\mathcal{D}(C_g, C_h)$ computed on row 4 of algorithm 3.1 immediately before C_g and C_h merge. Equivalently, two times $d_{coph}(\mathbf{x}_i, \mathbf{x}_j)$ is the path length from object $\mathbf{x}_i$ to object $\mathbf{x}_j$ on the respective dendrogram taxonomy tree. We call

$$\mathfrak{D}_{coph} = [d_{coph}(\mathbf{x}_i, \mathbf{x}_j)] \;\; i, j = 1, ..., n, \tag{3.3}$$

a *cophenetic matrix*. [14, p. 108]

(a) (b)

Figure 3.3: The single link (a) and the complete link (b) link criterion for the same two clusters of objects depicted in a two dimensional space. The line between the clusters shows the distance defined to the two clusters according to the two criteria.

3.1.3 Link criteria for clusters

It is clear that the row 4 distance function $\mathcal{D}(C_i, C_j)$ is the core of the algorithm 3.1 as it determines which two clusters merge at each iteration. We refer to this distance as the *link criterion* between two clusters. In chapter 2 we defined several distance measures for two single data objects. Link criteria extend this discussion to distances between sets of objects.

One of the simplest link criteria is the *single link* distance defined as

$$\mathcal{D}_{sl}(C_i, C_j) = \min_{\mathbf{x},\mathbf{y}}\{d(\mathbf{x}, \mathbf{y}) | \mathbf{x} \in C_i, \mathbf{y} \in C_j)\}, \tag{3.4}$$

where $d(\mathbf{x}, \mathbf{y})$ is the distance between $\mathbf{x}$ and $\mathbf{y}$. Note that function d can be chosen freely, for instance, any distance measure defined in chapter 2 can be used. Another used link criterion is the *complete link* distance

$$\mathcal{D}_{cl}(C_i, C_j) = \max_{\mathbf{x},\mathbf{y}}\{d(\mathbf{x}, \mathbf{y}) | \mathbf{x} \in C_i, \mathbf{y} \in C_j)\}. \tag{3.5}$$

Figure 3.3 gives an illustration of the two above-mentioned linkages.

As the distance function d determines how close the single data objects are in respect to each other, the linkage determines the shape of the arising clusters. Recall Figure 3.1. With single link clustering, single data objects tend to merge with existing clusters and from elongated shapes as opposed to forming new clusters. This is a phenomenon sometimes called as *chaining*. Complete link, on the other hand, tends to form more spherical clusters. [14, p. 111]

Single and complete link are two extreme methods of assigning distance between two clusters. In both, cluster distance is defined only according to a single distance between two objects in the clusters. A way to make the

merging criterion more balanced and less sensitive to outliers, is to utilize the *average link* distance

$$\mathcal{D}_a(C_i, C_j) = \text{average}\{d(\mathbf{x}, \mathbf{y}) | \mathbf{x} \in C_i, \mathbf{y} \in C_j)\}. \tag{3.6}$$

The average link is often referred as the unweighted pair group method average (UPGMA). This is because merging two clusters C_i and C_j, so that $|C_i|$ is essentially bigger, the behavior of the resulting cluster $C_i \cap C_j$ will be dominated by the bigger cluster C_i in the following steps of agglomerative clustering. In other word, nested clusters are not treated with equal weight in the procedure. If this is an undesirable property, we can use the *weighted average link* defined for clusters C_i, C_j and C_k as

$$\mathcal{D}_{wa}(C_k, C_i \cap C_j) = \frac{1}{2}(\mathcal{D}_{wa}(C_i, C_k) + \mathcal{D}_{wa}(C_j, C_k)). \tag{3.7}$$

It gives each cluster equal weight and is sometimes called as the weighted pair group method average (WPGMA).

A quality common to all the four link criteria mentioned above is *monotonicity*. This means that for any three clusters C_i, C_j and C_k

$$\mathcal{D}(C_k, C_i \cap C_j) \geq \mathcal{D}(C_i, C_j). \tag{3.8}$$

In other words the distances at which consecutive mergers happen increase monotonically.

3.2 Least squares tree searching

3.2.1 Optimal tree

It is known that the distance values of a cophenetic matrix produced by a monotonic agglomerative method satisfy the ultrametric property. Furthermore, if the original distance function over a set of objects is ultrametric to start with, than the respective cophenetic matrix will be exactly the original distance matrix. [30, p. 76,83]

Of course, in real life it is rare that a distance function computed over a set of data objects turns out to be ultrametric. For non-ultrametric distances, the fitting of a hierarchy can be seen as a way of forcing the distances to satisfy the ultrametric property. In this respect, if a hierarchical structure is present, the amount of forcing that we have to do, should be fairly small. Also, considering the process of taxonomy building, a chosen hierarchy should probably be the one for which the cophenetic matrix and the original matrix

differ the least. The problem with the clustering approach is that it does not directly optimize the overall fit, so with clustering we might end up altering the original distances more than we need to.

Perhaps, a better founded alternative is to think taxonomy building as an optimization problem in the search space defined by the taxonomy tree topologies and their respective branch lengths. The score we want to minimize, is then, for instance, the quantity

$$Q = \sum_{i=1}^{n} \sum_{j=1}^{n} \frac{(d(\mathbf{x}_i, \mathbf{x}_j) - d_{coph}(\mathbf{x}_i, \mathbf{x}_j))^2}{(d(\mathbf{x}_i, \mathbf{x}_j))^2}, \tag{3.9}$$

where $d(\mathbf{x}_i, \mathbf{x}_j)$ is the original distance between objects $\mathbf{x}_i$ and $\mathbf{x}_j$ and $d_{coph}(\mathbf{x}_i, \mathbf{x}_j)$ the corresponding fitted cophenetic distance in the inferred taxonomic hierarchy [1, 2, 19]. We call equation 3.9 the *Fitch-Margoliash least squares criterion*.

There is, is fact, an interesting relation between the ultrametric tree topology and least squares branch lengths. If we already have a tree topology defined, it has been shown that by assigning the merger heights according to the average distance between the objects in the merging subtrees, the resulting cophenetic distances will be least squares optimal. Proof for this can be found in [16]. Considering this result in the context of agglomerative clustering, this is basically what the average link does.

In this light, the average link is a fairly justified method, although the defect is that the space of different tree topologies is not thoroughly searched. From the optimization point of view, the optimal solution includes both best topology and best branch lengths. Because we know that the optimal branch length assignment can be done with the above mentioned average distance procedure, the remaining task is to find the right tree topology. The bad news is, however, that the search space of all tree topologies is overwhelmingly large. This means, for one, that exhaustive search of tree topologies is out of the question.

3.2.2 The Number of tree topologies

To get an idea of the tree search space size, we can examine the number of trees by imagining the building processes of all the trees with n leafs. Note that we are talking about rooted, labeled, binary trees. We can start from a tree with 1 leaf and count the number of alternative $n - 1$ size leaf branch adding sequences needed to construct all the trees with n leafs. If we constraint ourselfs to lexicographical adding order of labeled leaf branches,

number of leafs	5	10	15	20	30
number of trees	105	34 459 425	$\sim 2.1 \times 10^{14}$	$\sim 8.2 \times 10^{21}$	$\sim 5.0 \times 10^{38}$

Table 3.1: The number of rooted, bifurcated, labeled trees for different number of leaf nodes. [18, p. 24]

the number of adding sequences equals to the number of trees because a particular tree can be built only with a unique set of adding operations.

To elucidate the building process, consider adding the k:th leaf branch to a tree with $k-1 \geq 1$ leafs as the $k-1$ step of the building process. Altogether there are $(2k-3)$ edges in a tree with $k-1$ leafs, including the root as an edge. To get a tree with k leafs, we can add a leaf branch to any of the $(2k-3)$ edges on the tree. Respectively, the addition will increase the total amount of branches by two in the tree. Thus, for each consecutive step starting from a tree with 1 leaf, we have two additional places to add a leaf branch. From this we get that there are a total of

$$1 \times 3 \times 5 \times 7 \times 9 \times ... \times (2n-3) = \prod_{i=2}^{n}(2n-3) = \frac{(2n-3)!}{2^{n-1}(n-1)!} \qquad (3.10)$$

branch adding sequences in the construction process of all rooted, labeled, binary trees with n leafs. This, as we said, is also the size of the topology space for these trees. From table 3.1 it can be seen that the value of equation 3.10 grows rapidly as n grows. [18, p. 20-24]

3.2.3 Greedy tree inference

Of course, to find the optimal tree topology we can try more intelligent approaches than exhaustive search. For instance, we can apply the greedy traverse of the topology space, by trying different local subtree rearrangements and proceeding with the one for which the best branch length fit can be found.

A technique like this is the *Nearest-neighbor interchange* (NNI), where three subtrees of the main tree are disconnected by dissolving their connections to some interior branch. The three possible reconnection arrangements of the subtrees are then considered. From the three trees one will of course be the original tree. Figure 3.4 illustrates how this can be done. As a whole, an n leaf tree has $n-2$ interior branches, thus for one specific topology, $2(n-2)$ nearest-neighbor interchanges can be tried at each step of the optimization procedure. Various other greedy heuristics exist for local optimization of tree topology.

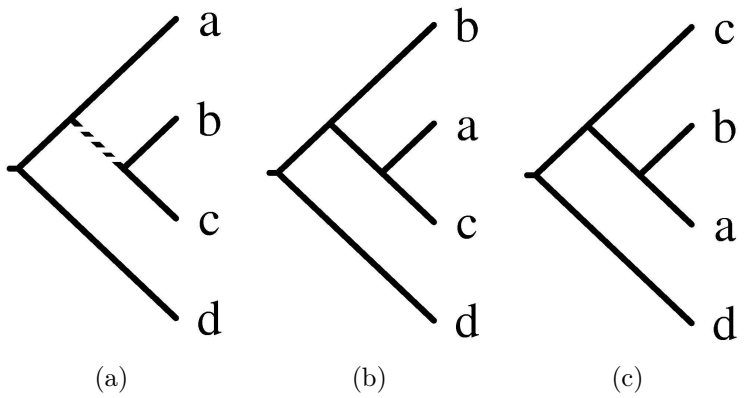


Figure 3.4: Nearest neighbor interchange. Three subtrees in Subfigure (a) are disconnected by dissolving the connections defined by the dashed interior branch. Subfigure (b) and (c) show the resulting alternative branch rearrangements.

To consider these methods in the context of dendrogram tree inference, we can start from an arbitrary full size tree topology and through rearrangement steps work our way towards the optima. However, a better strategy is to use the greedy approach already in the selection of the initial tree. We can start from a tree with 1 leaf and gradually build a tree by consecutively placing a leaf branch to the current tree so that equation 3.9 is minimized each time. This is called a *sequential adding strategy*. [18, p. 47-65]

Algorithm 3.2 defines a greedy tree building method incorporating the ideas we have discussed above. Like the agglomerative method in Subsection 3.1.2, the algorithm takes the set D of data objects as an input. Furthermore, the output is a taxonomic hierarchy, although following the discourse of this section we rather talk about a tree $\mathcal{T}_n(w)$ with n leafs and a branch length assignment w, then a hierarchy of nested clusterings. Still, from the view point of taxonomy building these are equivalent structures and can be presented as dendrograms.

Algorithm 3.2 starts by computing a distance matrix $\mathfrak{D}$ for the data objects. It then greedily builds a tree $\mathcal{T}_n$ with n leafs within n consecutive steps. For each step i, it fits a new labeled leaf branch to each of the $(2i-3)$ edges in $\mathcal{T}_{i-1}$ and picks the one that minimizes the Fitch-Margoliash least squares criterion Q after branch length assignment. Recall, that the optimal branch lengths w are obtain by assigning the merger heights according to the average distance between the objects in the merging subtrees [16]. The labeled lead branches are added in the order given in the input. After a full size tree is attained, nearest-neighbor interchanges are applied until the

Algorithm 3.2

Input: A data set $D = \{\mathbf{x}_1, ..., \mathbf{x}_n\}$.
Output: A taxonomic hierarchy tree $\mathcal{T}_n(w)$.

```
1: compute 𝔇_org = [d(x_i, x_j)]  i, j = 1, ..., n
2: initialize labeled tree T_0 with a root edge.
3: for each x_i ∈ D
4:    form T_i by assigning a new i label leaf branch
          to an edge e ∈ T_{i-1}, so that with optimal w,
             criterion Q(𝔇_org, 𝔇_coph(T_i(w))) is minimized.
5: end
6: while Q is reducible with NNI subject to an edge e ∈ T_n
7:    apply nearest-neighbor interchange to T_n
8: end
```

minima of Q in reached.

Not surprisingly, greedy strategies like this will, converge only towards a local optima. For example, the outcome of algorithm 3.2 depends on the order in which the labeled leaf branches are added to the initial tree. In fact, it has been shown that the problem of finding a globally optimal coupling of tree topology and branch lengths minimizing equation 3.9 is NP-hard [1]. In other words no polynomial algorithms are likely to exist for this problem.

3.2.4 Least squares vs. clustering

Although Algorithms 3.1 and 3.2 give no guarantees in terms of globally optimal least squares fit, for inference of ultrametric trees, both work fairly well in practice. Among the agglomerative clustering methods average link is the one with the best theoretical behavior and thus quite close to algorithm 3.2. Still, the framework around the greedy least squares method is, from the view point of taxonomy inference, more elegant and at least, guaranteed to converge to a local optima. On the other hand, average link is clearly less complex and invariant to input order and thus consistent in terms of output. In this respect, preference of one over the other is justified either way.

Chapter 4

Comparing and assessing taxonomic hierarchies

A different choice of distance measure and hierarchy building method, will probably result into a different inferred taxonomic hierarchy. All methods will, however, output some kind of hierarchy, should such fit a data or not. In this chapter we take a look at how distance between different output taxonomic hierarchies can be measured and how the fit of a ultrametric model can be assessed for a given data. These are important considerations and will help conclusion making when applying hierarchy methods to real data.

4.1 Methods for comparing taxonomic hierarchies

In chapter 1 we defined a taxonomic hierarchy as a rooted, labeled binary tree. In other words, when comparing two taxonomic hierarchies we are, in fact, comparing two tree structures. Fortunately, a variety of interesting methods has been developed for evaluating differences between trees. In the following subsections we will discuss a few. The compared trees are assumed to have the same set of leaf objects. For an additional overview on the subject one can refer among others to [5] and [18].

4.1.1 Comparison of path length

Perhaps the most obvious way of computing a distance between two labeled trees is to compare the distances they impose on their leaf object pairs. Recall that in a labeled tree, the distance $d(\mathbf{x}_i, \mathbf{x}_j)$ between two objects $\mathbf{x}_i$ and $\mathbf{x}_j$ is

the sum length of the branches that form a path from $\mathbf{x}_i$ to $\mathbf{x}_j$. Now say, $\mathcal{T}_1$ and $\mathcal{T}_2$ are the trees we want to compare and the functions d_1 and d_2 define the distances the trees impose on their leaf object pairs. The *path length difference distance* between the two trees $\mathcal{T}_1$ and $\mathcal{T}_2$ is [16, 42]

$$Q_\delta(\mathcal{T}_1, \mathcal{T}_2) = \sqrt{\sum_{i=1}^{n} \sum_{j=i+1}^{n} (d_1(\mathbf{x}_i, \mathbf{x}_j) - d_2(\mathbf{x}_i, \mathbf{x}_j))^2}. \tag{4.1}$$

Function Q_δ is, however, scale dependent. This means that, if the scale on branch lengths is different in the two trees, Q_δ deems $\mathcal{T}_1$ and $\mathcal{T}_2$ dissimilar, also in the case where the two trees are identical in topology. In addition, the values of Q_δ are unbound from above and thus poorly comparable in general terms. Hence, in stead of using the sum of squares we can use correlation. We define the *path length correlation distance* as

$$Q_{corr}(\mathcal{T}_1, \mathcal{T}_2) = corr(d_1, d_2), \tag{4.2}$$

where $corr(d_1, d_2)$ is the correlation between the distances imposed by the trees $\mathcal{T}_1$ and $\mathcal{T}_2$. $Q_{corr}(\mathcal{T}_1, \mathcal{T}_2)$ is scale invariant and bounded between -1 and 1.

4.1.2 Comparison of inner branch induced topologies

Instead of comparing the distances that two trees impose on their leaf object pairs, we can examine the differences in topology that the two trees have. Consider a tree $\mathcal{T}$ inferred form a set D of data objects. Consider also an internal branch e of the tree $\mathcal{T}$. With an internal branch we mean a branch not incident with any leaf of $\mathcal{T}$. By deleting e for $\mathcal{T}$, we divide the leaf objects into two subtrees of $\mathcal{T}$. This division of objects into two disjoint sets $A, B \subseteq D$, such that $D = A \cup B$, is called a *split induced* by e. Now, we can for instance, evaluate the difference of two trees by comparing the amount of similar splits their internal branches induce.

For convenience, topological study of trees is often restricted to *unrooted trees*. In an unrooted binary tree the degree of an inner node is always three and thus it is a simpler structure than a rooted tree. Figure 4.1 illustrates the difference between a rooted and an unrooted tree. The trees on the figure are considered unweighted, so the branch lengths should be discarded.

The simplest tree distance based on tree topology is the *symmetric difference*, also known as the *partition metric* [18, p. 529]. To compute the partition metric for two trees we go through the splits induced by each internal branch and count the number of splits that differ. Maximally this can

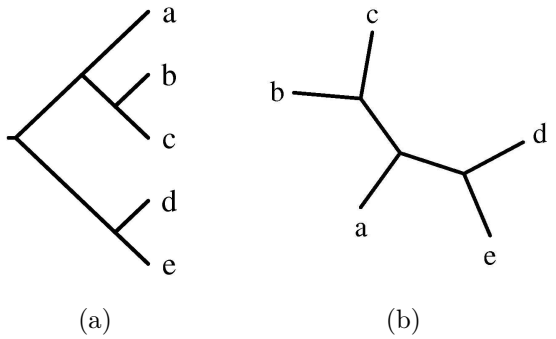


Figure 4.1: A depiction of a rooted tree (a) and the corresponding unrooted tree (b). The trees are considered unweighted, so the branch lengths should be discarded.

yield a value equaling two times the amount of inner nodes in the trees. For two taxonomic hierarchy trees $\mathcal{T}_1$ and $\mathcal{T}_2$ inferred from a set of data objects D, we define the partition metric as

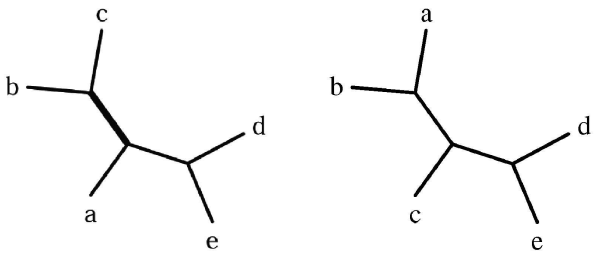
$$Q_{part}(\mathcal{T}_1, \mathcal{T}_2) = \frac{i(\mathcal{T}_1) + i(\mathcal{T}_2) - 2v_s(\mathcal{T}_1, \mathcal{T}_2)}{i(\mathcal{T}_1) + i(\mathcal{T}_2)}, \tag{4.3}$$

where $i(\mathcal{T})$ denotes the number of internal branches in $\mathcal{T}$ and $v_s(\mathcal{T}_1, \mathcal{T}_2)$ the number of pairs of identical splits on D induced by deleting an internal edge from each of $\mathcal{T}_1$ and $\mathcal{T}_2$ [46].

Example The split induced by the bolded branch of the tree in Figure 4.2(a) is $\{bc|ade\}$. As a whole, the tree in Figure 4.2(a) has splits $\{bc|ade\}$ and $\{abc|de\}$. The tree in Figure 4.2(b) has splits $\{ab|cde\}$ and $\{abc|de\}$. The split $\{abc|de\}$ is common for both trees. Hence, the value of the partition metric between the two trees is $(4 - 2 \times 1)/(2 + 2) = 0.5$

The partition metric has proven to be more sensitive to the differences and less sensitive to the similarities that the trees might have. That is, two somewhat differing trees that still have clear similarities, might get maximal distance value. In some situations such sensitivity to difference might be a good property but for a general purpose distance measure this is not very desirable. [41, 46]

An alternative distance measure that is more sensitive to partial similarities in trees is the *quartet distance* measure [12]. The idea of the distance is related to the partition metric. However, the quartet distance considers topologies of four leafs, instead of two way splits. Let $\mathcal{T}$ be a tree inferred



(a) (b)

Figure 4.2: Two five taxa trees with different topologies. The values for the partition metric and quartet distances between the trees are $\frac{2+2-2\times1}{2+2} = 0.5$ and $\frac{2\binom{5}{4}-2\times3}{2\binom{5}{4}} = 0.4$

from the set of data objects D. As already mentioned above an internal branch e of $\mathcal{T}$ splits the objects of D into two disjoint sets $A, B \subseteq D$, such that $D = A \cup B$. For any objects $\mathbf{x}_i, \mathbf{x}_j \in A$ and $\mathbf{y}_i, \mathbf{y}_j \in B$, we have the *quartet topology* $\{\mathbf{x}_i\mathbf{x}_j|\mathbf{y}_i\mathbf{y}_j\}$ and say the quartet topology is induced by e. Notice, that one internal branch induces multiple quartet topologies. To compute the quartet distance we go through, respectively, the quartet topologies induced by each internal branch and count the number of quartet topologies that differ. Maximally this can yield a value equaling two times $\binom{n}{4}$. We define the quartet distance for two taxonomic hierarchy trees $\mathcal{T}_1$ and $\mathcal{T}_2$ of n leafs as

$$Q_{quartet}(\mathcal{T}_1, \mathcal{T}_2) = \frac{2\binom{n}{4} - 2v_q(\mathcal{T}_1, \mathcal{T}_2)}{2\binom{n}{4}}, \tag{4.4}$$

where $v_q(\mathcal{T}_1, \mathcal{T}_2)$ denotes the number of pairs of identical quartet topologies induced by the internal edges of $\mathcal{T}_1$ and $\mathcal{T}_2$.

Example The quartet topologies induced by the bolded branch of the tree in Figure 4.2(a) are $\{bc|ad\}$, $\{bc|ae\}$ and $\{bc|de\}$. As a whole, the tree in Figure 4.2(a) has the quartet topologies $\{bc|ad\}$, $\{bc|ae\}$, $\{bc|de\}$, $\{ab|de\}$ and $\{ac|de\}$. The tree in Figure 4.2(b) has the quartet topologies $\{ab|cd\}$, $\{ab|ce\}$, $\{bc|de\}$, $\{ab|de\}$ and $\{ac|de\}$. The quartet topologies $\{bc|de\}$, $\{ab|de\}$ and $\{ac|de\}$ are in common for both trees. The quartet distance between the two trees is $(2\binom{5}{4} - 2 \times 3)/(2\binom{5}{4}) = 0.4$.

From the two above examples it is clear that quartet distance is much more complex than the partition metric, or the path length difference distance for that matter: the exhaustively comparison of all quartets one by one

takes $O(n^4)$ time as the number of topologies is $\binom{n}{4}$ in each tree. Surprisingly enough, the quartet distance can be computed in $O(n \log n)$ time [4]. Although granted that simplicity is the advantage for both partition metric and path length difference distance, quartet distance has been argued to have good properties for a general measure [46].

4.2 Validity of taxonomic hierarchies

4.2.1 Tests for measuring fit

When applying a hierarchy to a data set, it is always good to ask whether a hierarchical model is really justified for the data. This is especially important as building methods always return some kind of hierarchy whether it exists or not.

As we have already discussed in the previous chapters, a data has a "perfect" hierarchical structure only if it has a relevant ultrametric distance function for measuring similarity between the data objects. However, with real life data this is seldom the case. Hence, the matter of fit comes down to the question of how much do we have to alter the measured distances between the objects in order to achieve a hierarchical structure for the data.

One widely used test for fitting the ultrametric model, is the *cophenetic correlation coefficient* [15]. It is the correlation between the original distance and the distance in the cophenetic distance matrix of a given taxonomic hierarchy. For large values of the cophenetic correlation, the data can be considered to be hierarchically structured.

The problem is that defining what is a large correlation is not a straightforward question. Cophenetic correlation has proven to be quite sensitive in finding hierarchy also in random data [14, p. 109]. In other words, the value of cophenetic correlation should be very high before the original distance can be regarded as supportive of a hierarchical model. In general, however, the fundamental question of how large or small a fit test value should be applies to all fit tests equally. We will come back to this question in the next Subsection 4.2.2.

Another and perhaps a better fit measure for the ultrametric model is the *Goodman-Kruskal* γ statistic [20, 28]. It considers how the ordinal relationships between the distances change when a taxonomy is assumed. Hence, it is applicable when the rank order of a distance is more significant than the actual value. With rank we mean the index r of a distance d in a sorted list of distances between the objects.

To define the Goodman-Kruskal γ statistic, consider an $n \times n$ distance

matrix $\mathfrak{D} = [d(\mathbf{x}_i, \mathbf{x}_j)]$, $1 \leq i,j \leq n$, and the ranks r_k, $1 \leq k \leq (n^2-n)/2$, so that r_k corresponds to the rank of the k:th distance in the upper triangular of $\mathfrak{D}$, when enumareting from left to right and top to bottom. The Goodman-Kruskal γ statistic is defined as

$$\gamma = \frac{S_+ - S_-}{S_+ + S_-}, \tag{4.5}$$

where S_+ and S_- are the amount of *concordant* and *disconcordant* distance ranks between the original distance matrix and the cophenetic matrix. Concordant pairs for two $n \times n$ distance matrices $\mathfrak{D}^a$ and $\mathfrak{D}^b$ are all the doublets of rank pairs $\{(r_k^a, r_l^a), (r_k^b, r_l^b)\}$, $k<l$, that satisfy either $r_k^a < r_l^a$ and $r_k^b < r_l^b$ or $r_k^a > r_l^a$ and $r_k^b > r_l^b$. Respectively the doublets satisfying either $r_k^a < r_l^a$ and $r_k^b > r_l^b$ or $r_k^a < r_l^a$ and $r_k^b > r_l^b$ are disconcordant. A pair is neither concordant nor disconcordant if $r_k^a = r_l^a$ or $r_k^b = r_l^b$. Like the cophenetic correlation coefficient Goodman-Kruskal γ statistic gives values between -1 and 1.

Example Consider two distance matrices $\mathfrak{D}^a$ and $\mathfrak{D}^b$ and the corresponding distance rank matrices R^a and R^b so that

$$\mathfrak{D}^a = \begin{bmatrix} 0 & 0.7 & 0.9 \\ 0.7 & 0 & 0.3 \\ 0.9 & 0.3 & 0 \end{bmatrix}, \mathfrak{D}^b = \begin{bmatrix} 0 & 0.2 & 0.6 \\ 0.2 & 0 & 0.6 \\ 0.6 & 0.6 & 0 \end{bmatrix},$$

$$R^a = \begin{bmatrix} 0 & 2 & 3 \\ 2 & 0 & 1 \\ 3 & 1 & 0 \end{bmatrix} \text{ and } R^b = \begin{bmatrix} 0 & 1 & 2 \\ 1 & 0 & 2 \\ 2 & 2 & 0 \end{bmatrix}.$$

The upper right triangle entries of R^a and R^b are the rank sets $\{2, 3, 1\}$ and $\{1, 2, 2\}$. In the two sets we find one concordant doublet $\{(2, 3), (1, 2)\}$ and one disconcordant doublet $\{(2, 1), (1, 2)\}$. Hence, $S_+ = 1$, $S_- = 1$ and γ for D^a and D^b is 0.

4.2.2 Evaluating the fit of a taxonomic hierarchy

Fit measures like the Goodman-Kruskal γ statistic and the cophentic correlation give us a numerical value on how well a taxonomic hierarchy fits a given data. However, the most important question is still left open; how good must a good fit value be before it is reasonable to conclude a hierarchical structure?

Statistical test theory provides a good framework to approach the above mentioned problem [38]. The answer that is proposed to the goodness-of-fit

question is that a good fit must be significantly better than a usual fit of a non hierarchical dataset. What this means in practice is that we must compare the fit value obtained from the original dataset D to a distribution of the fit values obtained from a baseline population of datasets with no hierarchical structure. If only a very small proportion of the non-hierarchical data have a fit value better than D, then D can be though to have a reasonable good fit.

Of course, a baseline population might be hard to come by, but to elucidate the concept, let us assume that we know the distribution of the fit value of a baseline population. We denote the corresponding cumulative baseline distribution function with T. The idea is to assume a hypothesis that no hierarchical structure exists in the original data; that is, that our data belongs to the non-hierarchical baseline population as well. We call this the *null hypothesis* H_0.

Now, in order to come to the opposite conclusion that we actually do have hierarchical structure in the data, the null hypothesis must be falsified on the basis of evidence that we have in the data. Say, we get a value t from the fit test of the original data. The equation

$$P(T \geq t|H_0) = p \tag{4.6}$$

gives us the probability of the fit test value being originated from a sample of the baseline population, in other words from non-hierarchical data. If we are willing to belive that p is too small for this to be true, we can reject the null hypothesis and conclude hierarchical structure for the data. A reasonable p for falsification of H_0 might be for example 0.05 or under. We call p the *p-value* of the test.

4.2.3 Monte Carlo and the null model method

The p-value gives us a good feel of how strongly our fit supports the conclusion of hierarchical structure in the data. But still, to obtain a p-value we need the baseline population, or more specifically the distribution of their fit test values. Naturally, we do not have such, but we can approximate the distribution with a method called the *Monte Carlo analysis* [30, p. 143-148]

The name Monte Carlo comes from the roll of randomness incorporated in the method [35]. It was developed originally as a statistical approach to study differential equations in the context of particle physics. To illustrate the general idea, consider the problem of evaluating the volume of a 20-dimensional region r defined by a set of inequalities

$$f_1(x_1, ..., x_{20}) < 0; ...; f_{20}(x_1, ..., x_{20}) < 0. \tag{4.7}$$

Consider also that the region is located in a unit hyper cube and that the functions are computable, but their integrals are intractable. The approach taken by the Monte Carlo analysis is to randomly sample a certain amount of points from the hyper cube and approximate the volume of the region with the portion p of the sampled points satisfying the inequalities. With a sufficient amount of samples the estimate should be valid within a reasonably small interval. [36]

With taxonomies the Monte Carlo analysis is used similarly. Analogous to a region inside a hypercube, we assume a region $\hat{r}$ inside a space of non-hierarchical data sets. The region $\hat{r}$ is considered to consist of those data sets that have a better fit value then D. Furthermore, the surrounding 'hyper-cube' is defined with a formulation that captures the characteristics of the assumed non-hierarchical data sets. We call the formulation a *null model*. Now, again, we are interested in the proportion $\hat{p}$ of the datasets inside $\hat{r}$. The idea simply is to estimate $\hat{p}$ by constructing a sampling procedure that randomly generates dummy data sets from the null model data space, and see how many have a better fit value.

As an example, a null model might take the frequencies of the objects of D as a parameter. Other qualities of the dummy instance could be left to vary freely. Fixing the frequencies result to dummy data sets with objects having the same frequencies as the objects in D. Hence, fixing a parameter constant reduces the null model sampling space. On the contrary, letting the frequencies vary freely, the sampling space grows.

As the formulation of the null model defines the size of the sampling space it will also affect the p-value of the test. The fundamental question is therefore, what parameters of the model should be fixed and what should be left to vary? No general cook book instructions are available. The answer is dependent on the application domain, and the advice of the domain expert should be followed. Monte Carlo methods and null models have been applied among others in ecological research. There they have resulted into interesting findings but have also generated a lot of controversy. [21, p. 7-13]

4.2.4 Bootstrap method for evaluating the stability of a taxonomic hierarchy

The p-value we obtain from the Monte Carlo method is dependent on the parameters of the given null model. An inappropriate null model or false parameters can easily lead to wrong assumptions about the goodness of the inferred taxonomy. The *bootstrap* resampling method is a less parameter dependent way of assessing how much support a data set gives to some taxo-

nomical structure [18, p. 335-363]. It does not involve assessing ultrametricity. Instead, the bootstrap method evaluates the *stability* of the inferred taxonomy. With stability we mean the proportion of noise in the hierarchy.

The idea behind the bootstrap method is based on the notion that our set of data points $D = \{\mathbf{x}_1, .., \mathbf{x}_n\}$ is a finite sample taken from some unknown distribution F. By computing some property $\hat{\theta}$ from D, we are merely making an estimation of the true property θ of F. For instance, $\hat{\theta}$ can be the sample mean value. Now, if we knew F, we could see how good the estimate $\hat{\theta}$ really is by drawing a set of additional samples from F and examining how prevalent the quality $\hat{\theta}$ really is in F. More precisely, we could see how the estimates of θ vary when sampling F. What the bootstrap does is that it simulates the sampling procedure by resampling randomly n times with replacement the data points of D. In other words, it estimates F with the empirical distribution $\hat{F}$ of D, instead. A sample taken from $\hat{F}$ is called a *bootstrap replicate*, denoted D^*. For a large n and a sufficient amount of replicates, it is known that the variance of the quality $\hat{\theta}$ in the replicates will be similar in comparison to what we would get by sampling directly form F. [10]

In the context of taxonomy inference the property θ of interest is of course the taxonomy itself. In other words, we are interested in how much the taxonomies differ when sampled from F. If the variance of the features in the inferred replicate taxonomies is small, we can regard the hierarchical structure proposed by F as being stabile, that is having little random noise. Furthermore, if the taxonomy $\mathcal{T}$, which we have inferred from all available points in D, is close to the other replicate taxonomies, our certainty of $\mathcal{T}$ being supported by F is strengthened.

When θ is a quantitative value, say a mean for instance, we can just plot a histogram of the m replicate values $\hat{\theta}^*_1, .., \hat{\theta}^*_m$, and see how the estimates vary. For a mean we can easily calculate confidence boundaries as well [8]. With taxonomies this is more tricky as they are mathematically more complex objects. However, to obtain a set of quantitative values for a set of trees, one idea is to use a *reference tree*, and see how distance between the replicate trees and the reference tree behave [37]. From these distance we can then obtain a histogram. A reference tree could be, for instance, the tree inferred from all available points in D. If the reference tree is well founded, the bootstrap replicates should then distribute heavily near small distances. For further discussion on bootstrap and taxonomy trees refer to [11] and [18].

Chapter 5

Experimental results

In the previous chapters, we have dealt the topic of taxonomic hierarchy inference on a more general level. However, as we have stated, the motivation for our work comes from a practical starting point: we wish to experiment on a set of data consisting of occurrence ranges of European land mammals. Our interests have been in 1) finding out what kind of taxonomic hierarchies the occurrence of species in the data implies and 2) whether a hierarchical model is justified.

To address the research problem, we have applied the set of techniques discussed in chapters 2, 3 and 4. In the following we present the results. In Section 5.1 we give a more precise description of the mammal occurrence data and describe the preprocessing steps applied to the data before the experiments. In Section 5.2 we present the results of applying different distance measures to the occurrence of mammals in the data. Section 5.3 is concerned with the analysis of the hierarchies inferred based on the three best distance functions chosen in Section 5.2. It is here, in Section 5.3, where we collect our main evidence for answering the research problem. Section 5.4 summarize and concluded our studies.

5.1 Data and preprocessing

The applied data set is a collection of land mammal occurrence ranges in Europe. It is maintained by Societas Europaea Mammalogica [39]. The data is in grid form and it consists of 2670 roughly 40×40 km^2 large grid cells distributed across Europe. For each cell we have the information whether a species is present (1) or absent (0). The number of species is 194. Thus the data forms a 2670×194 binary matrix.

Before the experiments the data was preprocessed. As a first step, 68

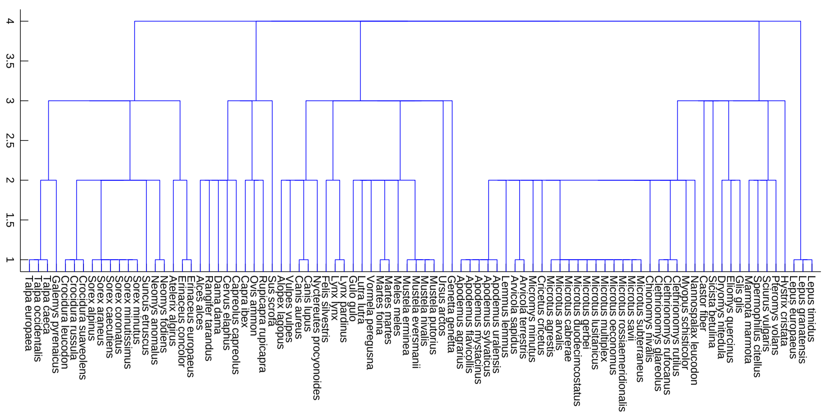


Figure 5.1: Dendrogram presenting the species of EMMA-data in Linnaean taxonomic order.

non-indigenous European species were removed. As a second step, all species having very low occurrence frequency (less that α=50 occurrences) were removed also. Finally, as a third step, all grid cells having less the β=8 species were taken out. The values of α and β were adjusted manually.

The main effect of the preprocessing was that clear outlier areas like Iceland and the Faroe Islands were left outside. After preprocessing, the remaining data had the dimensions of 2175 cells and 88 species. The collection of remaining species can be seen in Figure 5.1 in Linnaean taxonomic order. Table 5.1 shows the distribution of occurrence frequency for the remaining species. For future reference we refer to the 2175 $\times$ 88 data set as *EMMA-data*.

To connect climate variables to the cells of EMMA-data, we used the WORLDCLIM global climate data set distributed by the University of California, Berkeley [27]. The WORLDCLIM is also a grid data set, but with a cell resolution of roughly 18,5 $\times$ 18,5 km^2. It consists of weather measurements covering the whole world. The mapping between the two data sets was done, so that each EMMA-data cell was associate to the climate values of the closest WORLDCLIM cell in terms of coordinates. The variables used were the annual mean temperature, the temperature seasonality, the annual precipitation and the precipitation seasonality.

occurrence frequency class	0-20%	20-40%	41-60%	61-80%	81-100%
% of species occurring	51.2	20.5	5.7	17.1	5.7

Table 5.1: The occurrence frequencies of the species in the grid cells of EMMA-data.

5.2 Distance measures

In this section we present the results of applying different distance measures to the species of EMMA-data. All applied distance functions are based on grid cell occurrence of species. Our assessment approach has been to analyze the correlations and value distributions of the considered distance functions.

5.2.1 Internal measures

The value distribution for the considered internal distances between the species in EMMA-data are shown in the histogram plot of Figure 5.2.

Subfigure 5.2(a) depicts the histogram of the Hamming distance, denoted d_h. It is quite clearly distributed around three distance classes. The first spike at around 0.1 is generated by pairs of species that both have either high occurrence frequency ($>$0.4) or low occurrence frequency ($<$0.15). The third spike at around 0.65 is associated with species pairs that have a very differing occurrence frequency. Distances in the middle range of the histogram are assigned to pairs of species that both have medium occurrence frequency ($<$0.4 and $>$0.2). This suggested that the Hamming distance is quite affected by the frequency differences in the data.

For the Jaccard distance, denoted d_j, the histogram in Subfigure 5.2(b) shows that it's distribution is very heavily skewed towards the maximal value of one. Again, additional inspections showed that smaller Jaccard distances were assigned exclusively to pairs of high frequency species ($>$0.45). Pairs consisting of small or mixed frequency species were assigned a distance value close to 1. Hence, like the Hamming distance, the Jaccard distance seemed to be influenced by the large frequency differences in the data.

As Hamming and Jaccard distances showed sensitivity to frequency, other distance measures were considered. Alternatives for Jaccard were considered among the Second Kulczynksi, the Dice and the Cosine distances, denoted d_k, d_d and d_{cos}; all measures having the property of emphasizing similarity of occurrence as opposed to Jaccard that emphasizes difference [6].

The histograms of the Second Kulczynksi, the Dice and the Cosine distance are shown in subfigures 5.2(c), 5.2(d) and 5.2(e). In comparison to

Jaccard distance all three distance functions are able to even out the value distribution, although quite moderately. The smallest spike near values of 1 is generated with the Second Kulczynski. It also shows a small centralization values around 0.5 in the histogram. Recall the equation 2.5 in section 2.2.1: the Second Kulczynski will assign a value close to 0.5 for species pairs having a nested range and large frequency difference in occurrence.

As a further alternative, the correlation distance, denoted d_{corr}, was considered. It's histogram is shown in Subfigure 5.2(f). Two spikes are generated around values 1.1 and 0.9 indicating those species pairs that have a weak negative and positive correlation in occurrence. Still, the histogram is considerably more evenly distributed compared for instance to the histogram of the Second Kulczynksi distance.

Table 5.2 presents the correlation between the different considered distance functions. As expected the Jaccard type distances d_j d_k, d_d and d_{cos} are all strongly correlated among each other and weakly correlated with d_h. In addition, although d_{corr} is somewhat correlated with d_h, it is more correlated with Jaccard related functions. This is surprising because both d_{corr} and d_h react to co-occurring absence with increase of similarity unlike d_j d_k, d_d or d_{cos}.

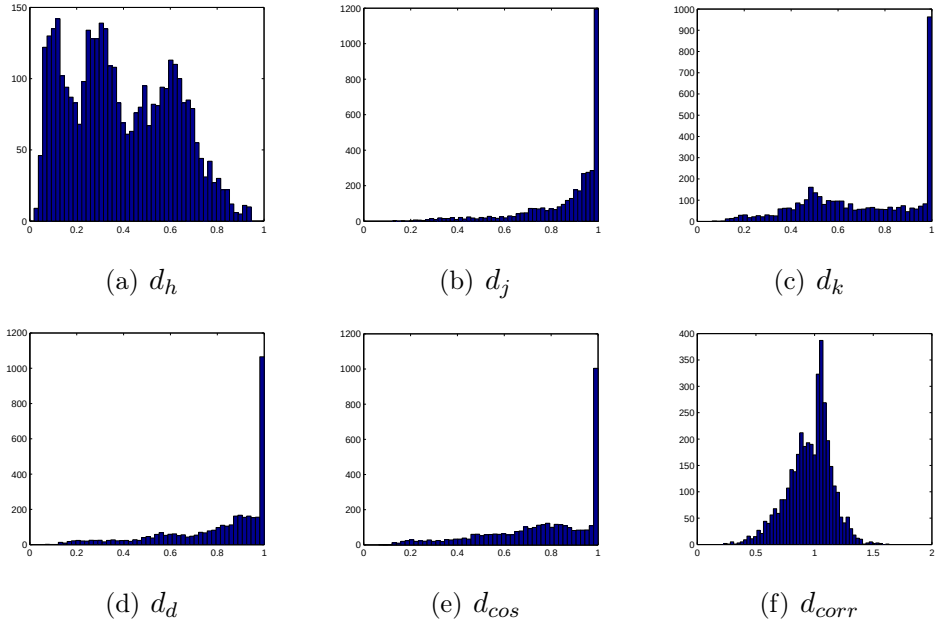
As a conclusion among the internal distances, the Correlation distance d_{corr} was considered most appealing: it's value distribution showed fairly good resolution power and it seemed to be the least affected by the large frequency differences of species occurrence. We chose d_{corr} to be applied in the next step of taxonomic hierarchy inference.

5.2.2 External measures

Our motivation for the external measures was based on an assumption that species niches related to ecology or something else would show in the data. The general idea being that species fighting for the same resources should occur in similar places, but rarely coexist. External measures should deem these kind of animals similar, whereas internal distances would not spot the similarity. For the assessment of external distance, Probe distance was considered best, because of its versatile and computationally affordable properties.

Figure 5.3 shows the value histograms of the probe distance functions considered. For simplicity, we refer to a probe distance composed of some (internal) distance function as the external version of that (internal) distance.

Subfigure 5.3(a) depicts the distribution of the external Jaccard distance, denoted $d^j_{p_a}$. The probe set has been assigned for $d^j_{p_a}$ as the entire set of species. Notice, that there are some similarities between the histogram 5.3(a) of $d^j_{p_a}$ and histogram 5.2(a) of d_h: both have a somewhat similar tri-modal



(a) d_h (b) d_j (c) d_k

(d) d_d (e) d_{cos} (f) d_{corr}

Figure 5.2: Histograms of distance values between species pairs in EMMA-data for each considered internal distance function. The Y-axis shows the number of pairs falling into a certain distance value threshold depicted on the X-axis. The distances denoted as follows: Hamming d_h, Jaccard d_j, Second Kulczynski d_k, Dice d_d, Cosine d_{cos} and Correlation d_{corr}.

	d_h	d_j	d_k	d_d	d_{cos}	d_{corr}	$d^j_{p_a}$	$d^k_{p_a}$	$d^{corr}_{p_a}$	$d^k_{p_{mf}}$	$d^{corr}_{p_{mf}}$
d_h	1.000	0.014	-0.325	-0.026	-0.133	0.264	0.847	0.784	0.406	0.608	0.2430
d_j	0.014	1.000	0.836	0.988	0.959	0.683	0.398	0.453	0.351	0.545	0.2790
d_k	-0.325	0.836	1.000	0.872	0.945	0.746	0.020	0.180	0.390	0.337	0.3850
d_d	-0.026	0.988	0.872	1.000	0.983	0.713	0.371	0.449	0.362	0.552	0.2940
d_{cos}	-0.133	0.959	0.945	0.983	1.000	0.749	0.254	0.368	0.382	0.495	0.3340
d_{corr}	0.264	0.683	0.746	0.713	0.749	1.000	0.410	0.591	0.744	0.640	0.6680
$d^j_{p_a}$	0.847	0.398	0.020	0.371	0.254	0.410	1.000	0.907	0.524	0.806	0.3090
$d^k_{p_a}$	0.784	0.453	0.180	0.449	0.368	0.591	0.907	1.000	0.641	0.933	0.4570
$d^{corr}_{p_a}$	0.406	0.351	0.390	0.362	0.382	0.744	0.524	0.641	1.000	0.669	0.8920
$d^k_{p_{mf}}$	0.608	0.545	0.337	0.552	0.495	0.640	0.806	0.933	0.669	1.000	0.5680
$d^{corr}_{p_{mf}}$	0.243	0.279	0.385	0.294	0.334	0.668	0.309	0.457	0.892	0.568	1.0000

Table 5.2: A table of correlations between distance functions for species pairs in EMMA-data. The internal distances denoted as follows: Hamming d_h, Jaccard d_j, Second Kulczynski d_k, Dice d_d, Cosine d_{cos} and Correlation d_{corr}. The external distances computed over the probe set of all species denoted as: external Jaccard $d^j_{p_a}$, external Second Kulczynski $d^k_{p_a}$ and external Correlation $d^{corr}_{p_a}$. The external distances computed over the probe set of middle frequency species denoted as: external Second Kulczynski $d^k_{p_{mf}}$ and external Correlation $d^{corr}_{p_{mf}}$.

distribution. Also the correlation value of Table 5.2 shows a high correlation between $d^j_{p_a}$ and d_h. This seemed surprising as internal Jaccard, d_j, itself is not correlated with d_h.

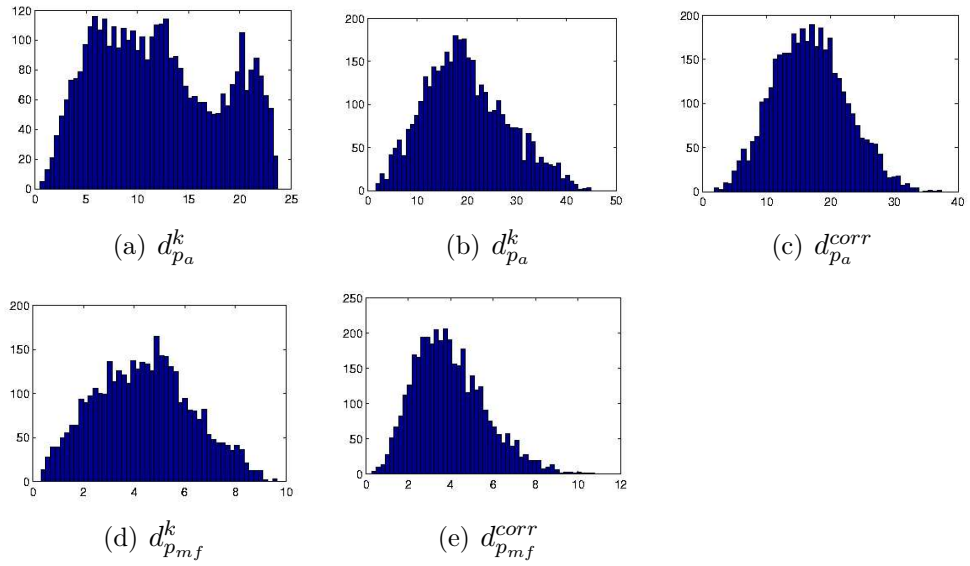
The explanation turned out to be, again, related to the large frequency differences in species occurrence. As we stated in the previous subsection, the smaller Jaccard distances were assigned exclusively to species pairs having both high frequency in occurrence. Pairs consisting of small or mixed frequency species were assigned a distance value close to 1. It is easy to see that pairs of high frequency species will thus behave similarly over the set of all probes. Respectively, because low frequency species are distant from everything else, probe comparison among these pairs will result to high similarity as well. From this it follows that pairs with a large frequency difference will behave differently in turn. Indeed, this results to frequency related behavior similar to what was detected with Hamming distance.

As an alternative for the Jaccard distance, the external measures of the Second Kulczynski and the Correlation distances were tested over the probe set of all species. After all, both d_k and d_{corr} showed some unique characteristics among the internal distance measures. We denote the corresponding external measures as $d^k_{p_a}$ and $d^{corr}_{p_a}$.

The resulting histograms of $d^k_{p_a}$ and $d^{corr}_{p_a}$ are shown in subfigures 5.3(b) and 5.3(c). What the subfigures show is a more normal-like value distribution in comparison to the histogram of $d^k_{p_a}$. Still, the same phenomenon of correlation with Hamming distance is seen with the $d^k_{p_a}$ and also to some extent with $d^{corr}_{p_a}$ (see Table 5.2). Furthermore, although the tendency is not as strong as with Jaccard distance, the external versions of the two measures were more correlated with Hamming distance than their internal counterparts.

Knowing that the large frequency differences were affecting the distance measures, an alternative probe sets of middle frequency (<0.2 and >0.4) species was considered. The idea was to avoid the pairwise comparison of species pairs with a very extreme occurrence frequency difference. The new probe set was then tried with the Second Kulczynski and the Correlation distance, denoted respectively as $d^k_{p_{mf}}$ and $d^{corr}_{p_{mf}}$.

Table 5.2 shows that the change to the middle frequency probe set did not result into a very dramatical difference when comparing the two external versions of the Second Kulczynski and Correlation distance; correlation for $d^k_{p_{mf}}$ and $d^k_{p_a}$ is 0.93 and for $d^{corr}_{p_a}$ and $d^{corr}_{p_{mf}}$ 0.89. However, correlation to the Hamming distance d_h is somewhat smaller with the middle frequency probe set than with the set of all species. Furthermore, Figure 5.3 shows slight chances in the value distribution. The histogram of $d^k_{p_{mf}}$ in Subfigure 5.3(d) is somewhat more evenly distributed than 5.3(b). The histogram of $d^{corr}_{p_{mf}}$ in



(a) $d^k_{p_a}$ (b) $d^k_{p_a}$ (c) $d^{corr}_{p_a}$

(d) $d^k_{p_{mf}}$ (e) $d^{corr}_{p_{mf}}$

Figure 5.3: Histograms of external distance values between species pairs in EMMA-data. The Y-axis shows the number of pairs falling into a certain distance value threshold depicted on the X-axis. The distances computed over the probe set of all species, denoted as: external Jaccard $d^j_{p_a}$, external Second Kulczynski $d^k_{p_a}$ and external Correlation $d^{corr}_{p_a}$. The distances computed over the probe set of middle frequency species denoted as: external Second Kulczynski $d^k_{p_{mf}}$ and external Correlation $d^{corr}_{p_{mf}}$

		Agl.clust.avgr.l.			Least squares		
		d_{corr}	$d^k_{p_{mf}}$	$d^{corr}_{p_{mf}}$	d_{corr}	$d^k_{p_{mf}}$	$d^{corr}_{p_{mf}}$
Agl.	d_{corr}	0	0.33	0.18	0.18	0.33	0.24
clust.	$d^k_{p_{mf}}$	0.33	0	0.33	0.36	0.12	0.36
avgr.l.	$d^{corr}_{p_{mf}}$	0.18	0.33	0	0.30	0.32	0.15
Least.	$dcorr$	0.18	0.36	0.30	0	0.35	0.29
Squares.	$d^k_{p_{mf}}$	0.33	0.12	0.32	0.35	0	0.33
	$d^{corr}_{p_{mf}}$	0.24	0.36	0.15	0.29	0.33	0

Table 5.3: Quartet distance values computed between the six considered dendrograms.

Subfigure 5.3(e) is by contrast more skewed towards smaller distance values compared to histogram 5.3(c).

As a conclusion, the behavior of the external distances seemed to be affected, as well, by the large frequency differences in the occurrence of species. However, from the five tested external distances the ones computed over the probe set of middle frequency species behaved most appealingly. We chose $d^k_{p_{mf}}$ and $d^{corr}_{p_{mf}}$ to be applied in the next step of taxonomic hierarchy inference.

5.3 Taxonomic hierarchies

Based on the discussion of the previous section we chose three distance functions d_{corr}, $d^k_{p_{mf}}$, and $d^{corr}_{p_{mf}}$ to be applied further. The distances were then utilized for taxonomic hierarchy building with both agglomerative clustering (algorithm 3.1) and greedy least squares method (algorithm 3.2). The used link criterion for clustering was average link. The least squares method was applied with ultrametric Fitch-Margoliash criterion, sequential adding and nearest-neighbor interchange. The program used for least squares inference was the Kitch routine of the Felsenstein PHYLIP[1] program package version 3.6 [17]. For clustering we used basic Matlab routines.

Two methods and three distance functions gave us all together six taxonomic hierarchy trees. For comparison between the inferred structures, Table 5.3 shows the set of quartet distances computed between the trees. The distance values indicate that the taxonomies are fairly similar, but not identical. Not surprisingly, trees produced with the same distance function but different inference method were the most closely related.

To avoid overlap in analysis caused by similar taxonomies, it was decided to reduce the number of trees by half in further studies. Since the differences were smaller in respect to the taxonomic hierarchy inference methods, the

[1] evolution.genetics.washington.edu/phylip.html

method/distance	d_{corr}	$d^{k}_{p_{mf}}$	$d^{corr}_{p_{mf}}$
Agl.clust.avrg.l.	0.55 (0.70)	0.66 (0.75)	0.65 (0.67)
Least squares	0.51 (0.73)	0.63 (0.73)	0.66 (0.71)

Table 5.4: Goodman-Kruskal γ and Cophenetic correlation (in parentheses) fit values for the six considered dendrograms.

reduction was done by selecting either the tree produced with the clustering method or the least squares method for each distance function. The fit values of the taxonomies were used as choosing criterion for qualitative analysis. The taxonomic hierarchy showing a better fit to the original distance function was kept for further analysis. In the computationally intensive quantitative analysis (Monte Carlo and Bootstrap) we used the average link agglomerative method for all three distance functions.

Table 5.4 shows the Goodman-Kruskal γ and cophenetic correlation fit values computed for the six taxonomies. For $d^{k}_{p_{mf}}$ the average link agglomerative clustering produced a better fit with both considered fit values, whereas least squares gave a better fit for $d^{corr}_{p_{mf}}$. For d_{corr} the situation was more ambivalent as the agglomerative method gave a better γ value, although the least squares method was better in terms of cophenetic correlation. For qualitative analysis, we ended up in favor of the agglomerative method for d_{corr}.

The dendrograms of the three remaining taxonomies are shown in figures 5.4, 5.5 and 5.6. A more close visual inspection confirms, indeed, that the dendrograms are different but have a lot of similar structure; the partitions of the species by the top most nodes are fairly similar whereas the lower nodes have slight differences. A more illustrative analysis will be presented in the next subsection.

5.3.1 Qualitative assessment

To analyze the hierarchical structure of species present in the dendrograms, we considered biology, ecology and geography related options. From these three, the alternative of biology associated structure was first ruled out. Also no indication of ecology related structure was found. By contrast, the analysis suggested that the produced dendrograms are related to geographical occurrence of species. For example a fairly clear division of northern and southern species is present already at the top nodes of the dendrograms. Going a bit deeper in the nodes of the trees more distinct groupings of species are present in terms of geographical areas, like the Alps, Mediterranean and Scandinavia.

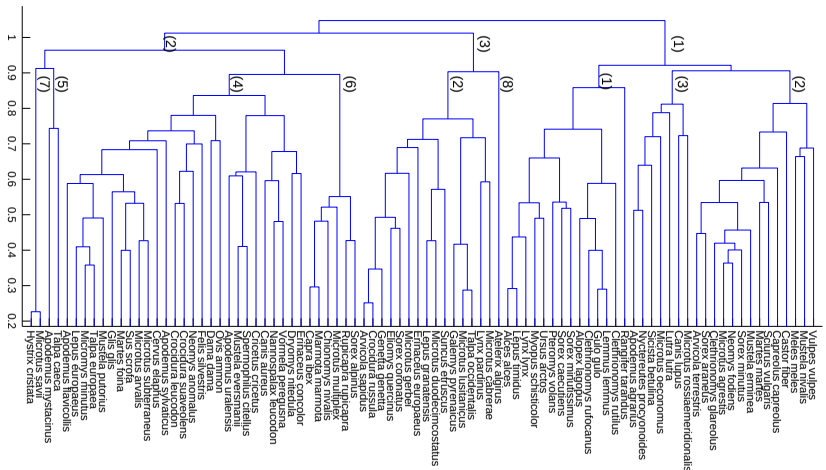


Figure 5.4: Dendrogram produced with d_{corr} and average link agglomerative clustering. The numbers in parentheses are related to the cluster indices of Figure 5.7

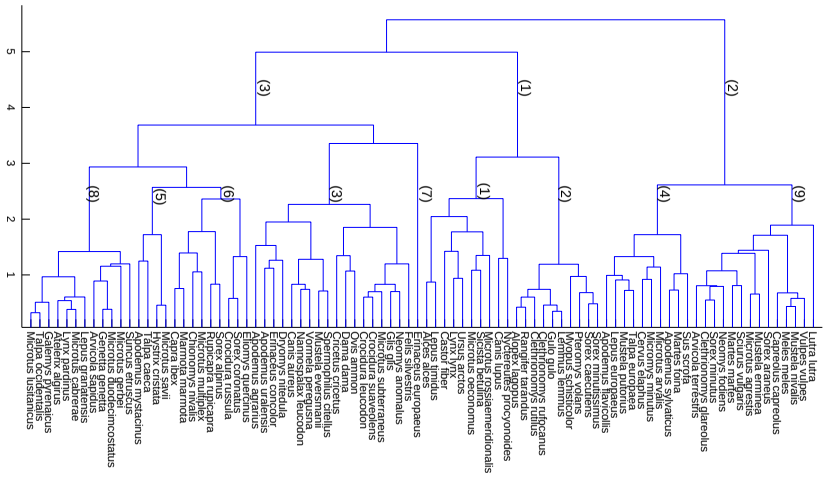


Figure 5.5: Dendrogram produced with $d^k_{p_{mf}}$ and average link agglomerative clustering. The numbers in parentheses are related to the cluster indices of Figure 5.8

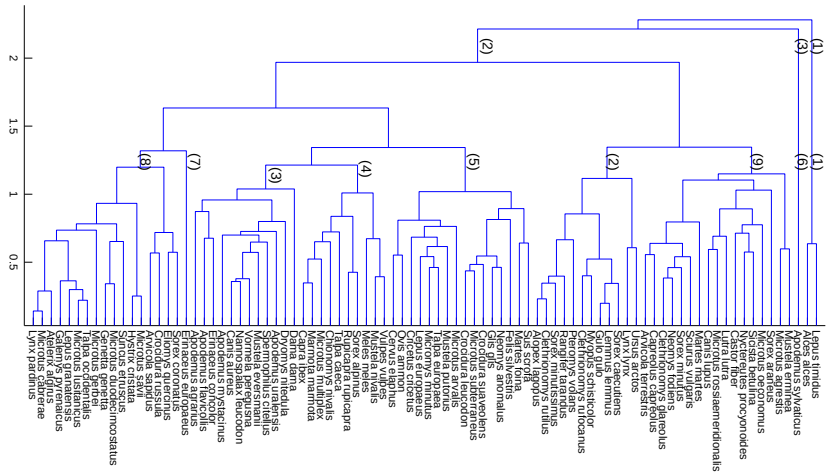


Figure 5.6: Dendrogram produced with $d^{corr}_{p_{mf}}$ and least squares method. The numbers in parentheses are related to the cluster indices of Figure 5.9

To study this further we plotted the joint occurrences of species in clusters defined by the dendrograms at two different levels of hierarchy. The levels divide the species into three and nine cluster partitions. The clusters are marked in figures 5.4, 5.5 and 5.6 with little index numbers.

Figures 5.7, 5.8 and 5.9 show the resulting joint ranges of a the species in each cluster. In each figures one cluster is represented by one small grid cell plot of Europe. The subfigures (a) show the partitioning into three clusters and subfigures (b) show the partitioning into nine clusters. The coloring of the small maps is assigned according to the amount of species occurring in a specific region in proportion to the total number of animals in the corresponding cluster. Thus, if the occurrence of the species of a certain cluster is concentrated, say, to northern Europe, than the corresponding map will have a red coloring in the north and a blue in south. The used color range is the spectrum from red to blue, so that red indicates presence of almost all species of the cluster and blue respectively dominant absence.

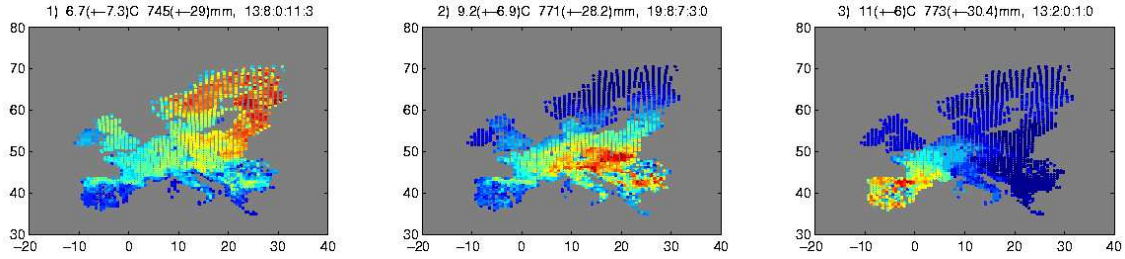
The numbers above each map indicate the following. The first is the index of the cluster. We denoted it in parentheses in further text. The second group of numbers are the mean and the seasonality of temperature (C) and the rainfall (mm) in the clusters. The values are cell averages from the WORLDCLIM data set, so that the average is weighted for each cell with the occurrence of the number of species in the cluster. The remaining numbers show the distribution of species in the clusters falling into the five occur-

rence frequencies classes (0-20%,20-40%,41-60%,61-80%,81-100%) analogous to Table 5.1.

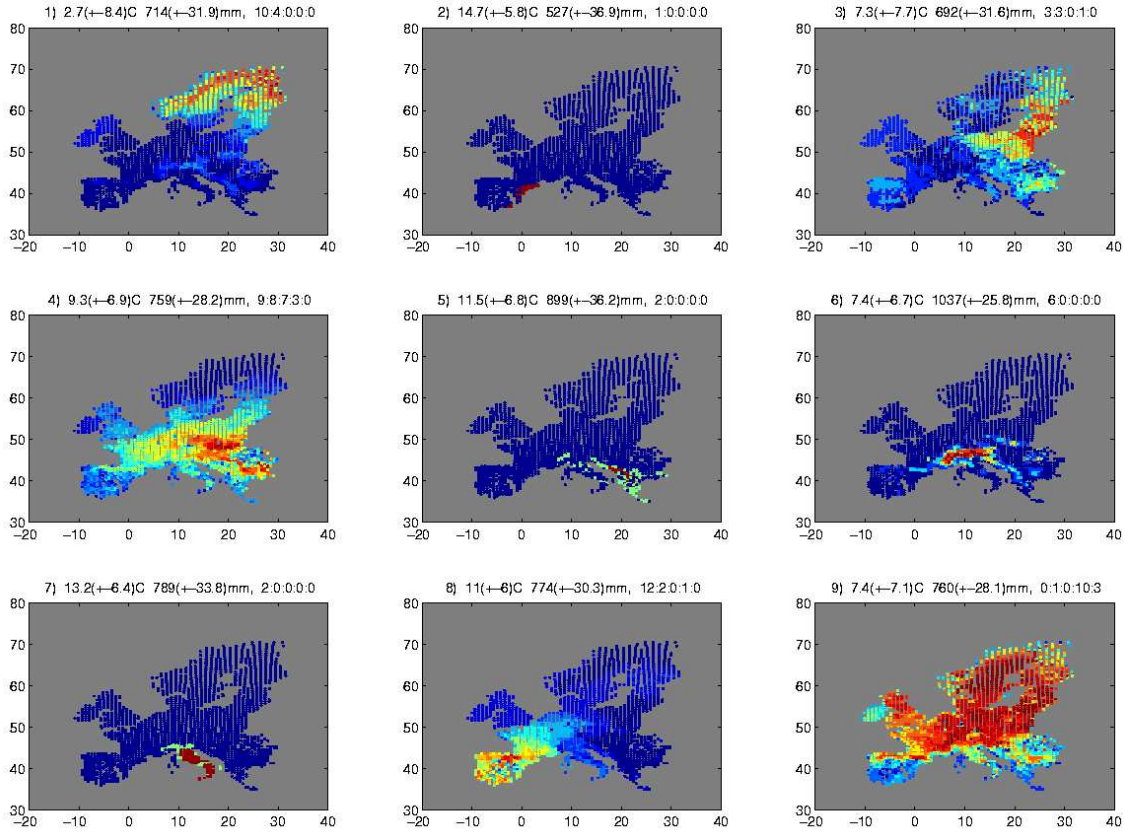
We start with Figure 5.7. It shows the division of species occurrence defined by the d_{corr} dendrogram of Figure 5.4. Subfigure 5.7(a) shows that the dendrogram partitions the species ranges nicely to a northern cluster (1), a central-eastern cluster (2) and a south-western cluster (3). On the level of nine clusters (Subfigure 5.7(b)), the northern cluster breaks into a Fennoscandic (1), a Baltic (3) and a pan-European cluster (9). The pan-European cluster keeps most of the high frequency species of the northern cluster, while the formed Fennoscandic and Baltic clusters are left with smaller ranged local species. The central-eastern cluster breaks, in turn, into a smaller central-eastern cluster (4), an alpine cluster (6) and two 2 species Mediterranean clusters (5) and (7) consisting of the species Talpa caeca and Apodemus mystanicus (5) and Microtus savii and hystrix cristata (7). All the high frequency species are left in the central-eastern cluster (4). The south-western cluster stays practically the same on the two hierarchy levels of the figure. The only species splitting from the south-western cluster is Atelerix algirus.

Moving on to Figure 5.8, we see the division of species made by the $d^k_{p_{mf}}$ dendrogram of Figure 5.5. The overall trends in the figure are similar to Figure 5.7, although some differences exist. On the level of three clusters we have again a northern (1) and a central European (2) cluster. However, the third cluster (3) is spread around central Europe, but appears to be more concentrated around mountain areas like the Pyrenees, the Alps and the Carpathians. On the level of nine clusters the northern cluster splits into a Fennoscandic (2) and a Baltic (1) cluster. The central European cluster splits two ways as well, into a continental central European cluster (4) and a more north spread pan-European cluster (9). The second, mountain area concentrate, central European cluster breaks into an eastern central European cluster (3), an eastern Mediterranean cluster (5), an alpine cluster(6) and a south-western cluster (8). The one species cluster (7) consisting of Erinaceus europaeus (hedgehog), splits also from the second central European cluster. Looking at the division of species frequencies classes in the clusters of Figure 5.8, no bimodality is present. All of the clusters consist to a large extent of either low or high frequency species, even at the three cluster level. This is a difference to the clusters of Figure 5.7(a).

Finally, Figure 5.9 shows the partition of species defined by the $d^{corr}_{p_{mf}}$ dendrogram of Figure 5.5. Already at the level of three clusters we have three species deemed as outliers by the method. These species are Lepus timidus (the Blue Hare) and Alces alces (the Elk) in cluster indexed (1) and Apodemus sylvaticus (the Wood mouse) in cluster indexed (3). Cluster indexed (2) contains all other animals in the EMMA-set. On the nine cluster

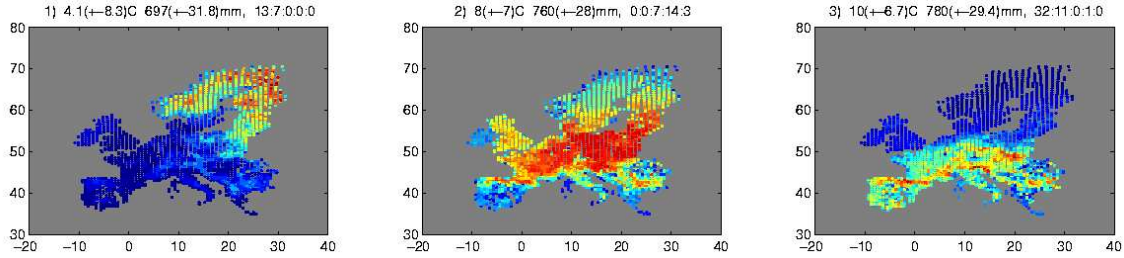


(a) The clusters at the level of three categories.

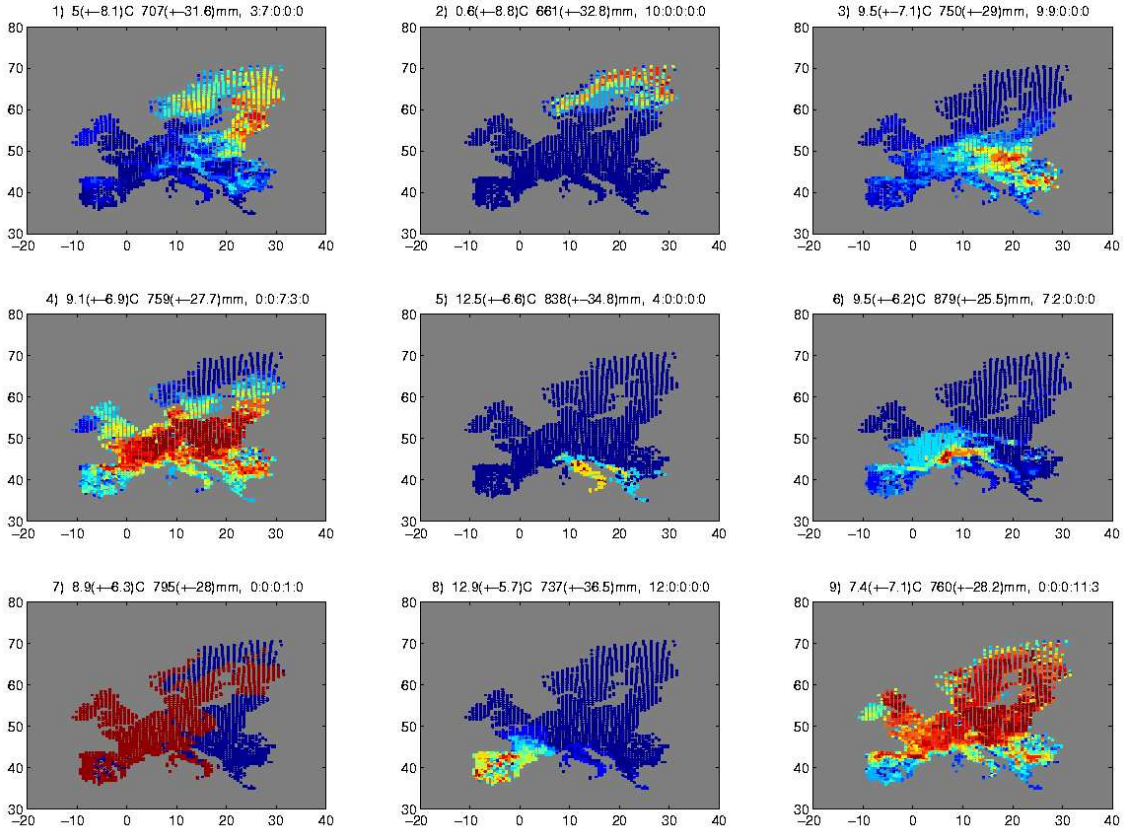


(b) The clusters at the level of nine categories.

Figure 5.7: Co-occurrences of species in clusters defined by two different levels of hierarchy in the d_{corr} dendrogram of Figure 5.4 inferred with average link agglomerative clustering. Each map shows the percentage of the cluster's species that occur in each cell of the grid. Warm colors indicate a high percentage and cold colors indicate a low percentage. The index of the cluster, mean cluster temperature and rainfall and the distribution of occurrence frequency classes of the cluster's species are market to the upper edge of each small map.

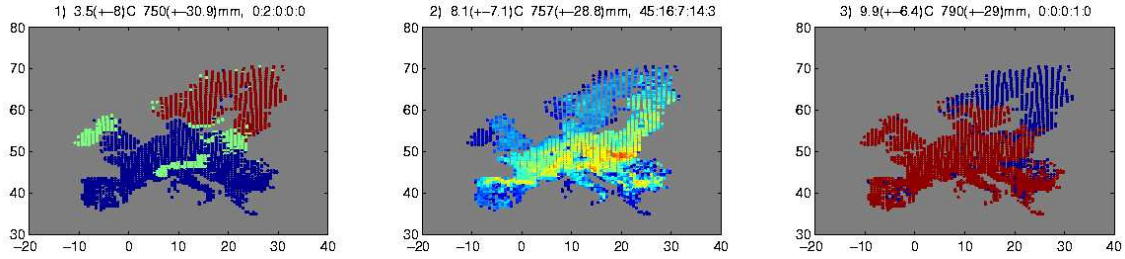


(a) The clusters at the level of three categories

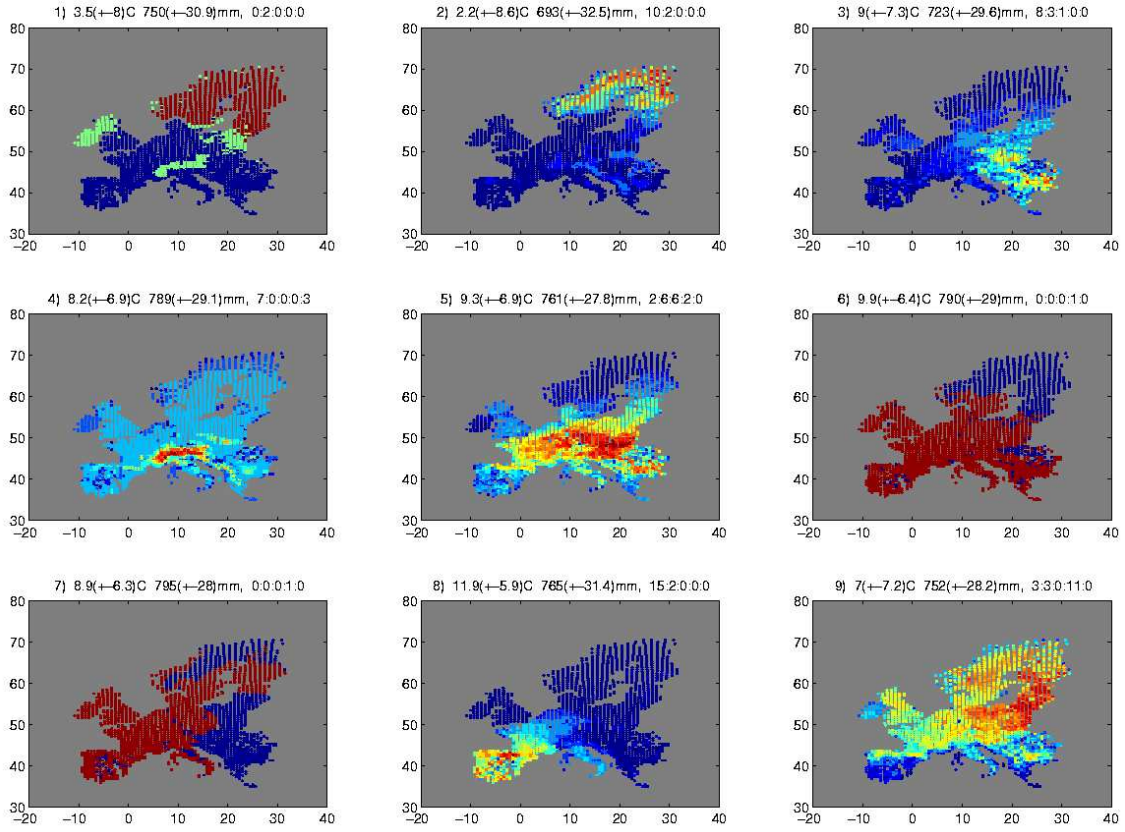


(b) The clusters at the level of nine categories

Figure 5.8: Co-occurrences of species in clusters defined by two different levels of hierarchy in the $d^k_{p_{mf}}$ dendrogram of Figure 5.5 inferred with average link agglomerative clustering. Each map shows the percentage of the cluster's species that occur in each cell of the grid. Warm colors indicate a high percentage and cold colors indicate a low percentage. The index of the cluster, mean cluster temperature and rainfall and the distribution of occurrence frequency classes of the cluster's species are market to the upper edge of each small map.



(a) The clusters at the level of three categories



(b) The clusters at the level of nine categories

Figure 5.9: Co-occurrences of species in clusters defined by two different levels of hierarchy in the $d^{corr}_{p_{mf}}$ dendrogram of Figure 5.5 inferred with the heuristic least squares method. Each map shows the percentage of the cluster's species that occur in each cell of the grid. Warm colors indicate a high percentage and cold colors indicate a low percentage. The index of the cluster, mean cluster temperature and rainfall and the distribution of occurrence frequency classes of the cluster's species are market to the upper edge of each small map.

p-value for $dim = 3$	$t = 0.5$	$t = 0.6$	$t = 0.7$	$t = 0.8$	$t = 0.9$
Subfigure 5.7(b)	$\sim$0	$\sim$0	0.22×10^{-12}	0.0014	0.46
Subfigure 5.8(b)	$\sim$0	0.34×10^{-14}	$0.25\ \times10^{-9}$	0.0018	0.074
Subfigure 5.9(b)	$\sim$0	0.33×10^{-8}	0.00002	0.0082	0.18

Table 5.5: The p-values of the MANOVA test on cluster climate profiles with different values of t. The test is done at the level of nine clusters of the dendrograms shown in figures 5.7(b) - 5.9(b). The p-values are for the hypotheses that the cluster means for four WORLDCLIM climate variables span a tree dimensional space (dim = 3). p-values for hypotheses of dim < 3 are not shown, but are all $\sim$0.

level only the large cluster splits. The split results to a set of clusters familiar from the previous figures 5.7 and 5.8. Once again, we have the Fennoscandic cluster (2), the Alpine cluster (4), the continental central European cluster (5), the south-western cluster (8) and the Baltic cluster (9). The one species cluster (7) consisting of Erinaceus europaeus (the hedgehog) is present here as well. The eastern central European cluster of Figure 5.8(b) is here however a little bit lighter and more south-east concentrated. Bimodality of species frequencies is noticeable here in the Alpine cluster and the Baltic cluster even at the level of nine clusters unlike on figures 5.7 and 5.8.

Reassurance for cluster structure

The structure seen in figures 5.7 - 5.9 clearly suggested a geographical division of species. However, considering that we defined similarity between species based on grid cell occurrence, a clustering reflecting different geographical areas could have emerged in any case. To get reassurance that the cluster regions are indeed reasonable and not of arbitrary origin, we wanted to test whether they are natural also in terms of climate qualities measured independently from the species.

For validation of the clusters in terms of climate, we assigned, for each cluster, a set of climate values from the WORLDCLIM data set (annual mean temperature, the temperature seasonality, the annual precipitation and the precipitation seasonality). More precisely, the set of climate values for each cluster was defined according to those grid cells that had a large proportion of the clusters species present: let s_i^C be the number of species belonging to a certain cluster C in the i:th cell of EMMA-data. The climate values of the i:th cell were associated to cluster C, if $\frac{s_i^C}{|C|} \geq t$, where t is a threshold between 0 and 1.

For the set of climate values, a one-way multivariate analysis of variance

(MANOVA) was carried out on the level of nine clusters [43, p. 156]. Table 5.5 shows the results. The MANOVA analyses suggested that the means of nine clusters respect to climate values are significantly differing and that the group means span a four dimension space. Since we have four climate variables, this is the largest possible dimension. The p-values for the hypotheses that cluster means span at least a three dimensions space are significant for all tests with $t < 0.9$. With significance, we mean a p-value smaller than 0.05. The only not significant test result is with $t = 0.9$. This is due to the Fennoscandic cluster missing from the test; the condition $t = 0.9$ is too strict for the Fennoscandic climet values to be included in the analysis. All in all the results of the MANOVA support clearly the geographical division defined by the clusters shown on subfigures 5.7(b), 5.8(b) and 5.9(b).

5.3.2 Quantitative assessment

Monte Carlo analysis

To test whether the fit values of Table 5.4 are better then what we would get from non-hierarchical data, a Monte Carlo analysis was conducted. This was done to answer the question of whether conclusion of hierarchical structure is justifiable based on any of the three distance measures used in the taxonomies inference.

To generate a non-hierarchical baseline population, a generative null-model was constructed . The idea of the null-model being that the occurrence ranges of the species are formed independently of each other. Only random structure of nested clustering should therefore arise from such a model. Thus, the overall null-model consisted of a set of independent sub models assigned one per each species.

In general, the occurrence patterns of the species in EMMA-data consist of one or several geographically uniform areas. To get a credible base-line population we wanted to incorporate this to the null-model of species occurrence. Hence, a parameter controlling the geographical fragmentariness of species range was added to the submodels.

To apply geographically uniform areas to the random model, a graph representation of each species occurrence range was formulated, so that cells having the species present were assign as vertices. Edges were respectively assigned between two vertices if and only if their corresponding cells were neighboring grid cells in the directions of north, south, east or west. Thus a cell-vertex was assigned to have maximally one neighbor vertex in each directions of north, south, east and west. Cells along a coastline were assigned accordingly less than four edges. We call this the species' *habitat graph*. The

Algorithm 5.1

Input: A set $C = \{c_1, ..., c_n\}$ of integers.
Output: A set $\hat{S}_k \subseteq G$ of grid cells.

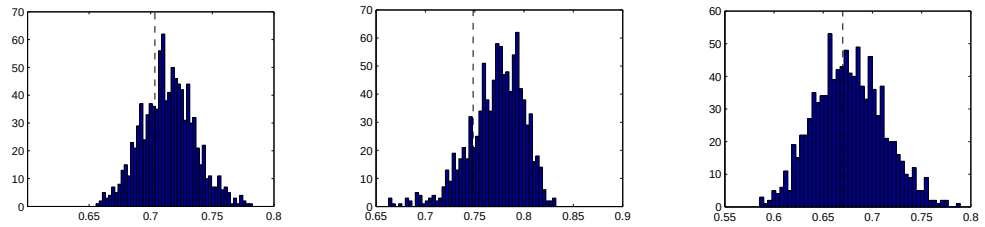
```
1: Ŝ_k = ∅
2: for each c_i ∈ C
3:   randomly assign v | v ∈ G and v ∉ Ŝ_k
4:   j = 1, N = v.neighbors
5:   while j ≤ c_i and N ≠ ∅
6:      Ŝ_k = Ŝ_k ∪ v
7:      N = N ∪ {v.neighbors \ {Ŝ_k ∩ v.neighbors}}
8:      randomly assign v | v ∈ N
9:      N = N \ v
10:      j = j + 1
11:   end;
12: end;
```

geographical fragmentariness of a species range was hence considered as the number and sizes of the connected components in the species' habitat graph in the EMMA data.

Now, let us denote G as the set of grid cells in EMMA data and let $S_k \subseteq G$ be the set of grid cells where the k:th species is present. Algorithm 5.1 defines the randomized generation process of one occurrence pattern $\hat{S}_k \subseteq G$ having similar geographical fragmentariness to S_k. Thus, to produce one randomized null-model data entry, the algorithm is ran once for each species S_k in the EMMA-data. The algorithm takes as an input parameter a set of integers $C = \{c_1, .., c_n\}$, where n is the number of connected components of the habitat graph for species S_k and an entry $c_i \in C$ is the size of the i:th connected component. The algorithm adopts ideas form [26].

Algorithm 5.1 runs as follows. For each connected component in species' S_k habitat graph, it randomly selects a new starting cell $v \in G$ as a starting point for a new random uniform area in $\hat{S}_k$. Furthermore, each area is initialized by assigning the neighbors of the start cell v as the initial set N of border cells of the uniform area. The area is then generated with an accumulation process by expanding at each step randomly the current area cells with a border cell belonging to N and updating N accordingly. The algorithm accumulates the current cell area randomly until the size defined by c_i is met or $N = \emptyset$.

The process produces a $\hat{S}_k$ with a similar geographical fragmentariness



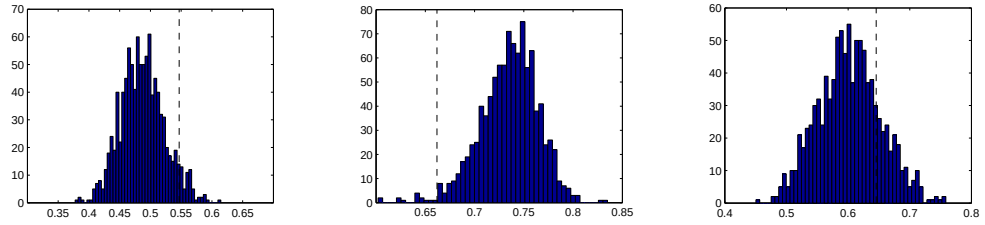
(a) d_{corr}:P(T≥0.70|H_0)=0.69 (b) $d^k_{p_{mf}}$:P(T≥ 0.75|H_0)=0.81 (c) $d^{corr}_{p_{mf}}$:P(T ≥ 0.67|H_0)=0.56

Figure 5.10: Monte Carlo analysis. The p-values and the corresponding baseline distributions of cophenetic correlation for the taxonomies inferred with average link agglomerative clustering and three different distances functions.

and size to S_k. However, it must be pointed out that this does not hold strictly. The number of uniform areas can be smaller in $\hat{S}_k$ if two or more random areas merge at some point of the random accumulation process. Also, the accumulation of the occurrence range can end up in a dead end if the current accumulated uniform area gets stuck in the middle of a coastline and another connected area. In this case N becomes empty and the size defined by c_i is not met. This leaves $\hat{S}_k$ smaller the S_k. However, in practice for EMMA-data the size of $\hat{S}_k$ followed the size of S_k fairly well.

The Monte Carlo analysis was conduced by generating a baseline population of 1000 randomized datasets of 88 species ranges according to algorithm 5.1 [34]. From each randomized data set a taxonomic hierarchy was inferred with the d_{corr}, $d^k_{p_{mf}}$ and $d^{corr}_{p_{mf}}$ distance functions and average link agglomerative clustering. Fit values were then computed for the baseline taxonomies. These being the cophentic correlation and the Goodman-Kruskal γ statistic. Finally the baseline fit values were compared to the corresponding fit values of the EMMA-data taxonomies.

Figures 5.10 and 5.11 show the results of the Monte Carlo analysis. When measuring fit with cophenetic correlation (Figure 5.10) none of the three distances d_{corr}, $d^k_{p_{mf}}$ or $d^{corr}_{p_{mf}}$ produce a significantly better fit that what the non-hierarchical model would suggest; significance meaning a p-value smaller than 0.05. Noticeable is, however, that the fit for $d^k_{p_{mf}}$ is worse than 82 % of the instances of the non-hierarchical model. In turn, measuring fit with the Goodman-Kruskal γ statistic (Figure 5.11) we get a nearly significant fit for d_{corr}. On the contrary, for $d^k_{p_{mf}}$ the fit is significantly worse than what the non-hierarchical model would suggest. The fit for $d^{corr}_{p_{mf}}$ is fairly good but not significant. In summary, the results of the Monte Carlo analysis gave reasonable support for the validity of a hierarchical model only for d_{corr}.



(a) d_{corr}:P(T≥0.55|H_0)=0.06 (b) $d^k_{p_{mf}}$:P(T≥0.66|H_0)=0.98 (c) $d^{corr}_{p_{mf}}$:P(T≥0.65|H_0)=0.22

Figure 5.11: Monte Carlo analysis. The p-values and the corresponding baseline distributions of the Goodman-Kruskal γ statistic for the taxonomies inferred with average link agglomerative clustering and three different distances functions.

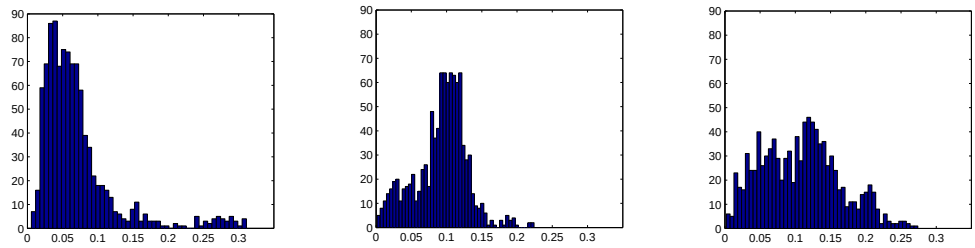
Bootstrap

To compare the stability of the inferred taxonomies a simple bootstrap analysis was conducted. For the analysis 1000 bootstrap replicates were taken. This was done by taking samples from the rows (cells) of EMMA-data. From each bootstrap replicate a taxonomic hierarchy was inferred using d_{corr}, $d^k_{p_{mf}}$ and $d^{corr}_{p_{mf}}$ and average link agglomerative clustering. The replicate taxonomies were then compared to a corresponding reference tree inferred with the same distance function but from all available data. For d_{corr} and $d^k_{p_{mf}}$ the reference trees were the taxonomies shown in the in figures 5.4 and 5.5. For $d^{corr}_{p_{mf}}$ the reference tree was produced with average link agglomerative clustering and therefor it was not the dendrogram of Figure 5.6. The comparison between the replicates and the reference tree was done using quartet distance.

The results are depicted in Figure 5.12. They show that on the average the distance between the replicate taxonomies and the reference tree is the smallest with d_{corr} and largest with $d^{corr}_{p_{mf}}$. By contrast, the variance is smallest with $d^k_{p_{mf}}$, whereas the variance of d_{corr} is increased by the long tail of it's histogram. Still, clearly the distribution of d_{corr} is more compact and peaked than what $d^k_{p_{mf}}$ or $d^{corr}_{p_{mf}}$ are. This is supported by the *kurtosis* value computed for of the histograms. We defined kurtosis κ by the formula $\kappa = \frac{\mu^4}{\sigma^4}$, where μ^4 is the fourth central moment and σ^4 the fourth power of the standard deviation of the values in the histogram.

5.4 Conclusions

The aim of this study was to address two question: 1) What kind of structure does the occurrence of species in the data imply? 2) Is an ultrametric hier-



(a) d_{corr} μ=0.071 σ^2=0.0030 κ=9.18 (b) $d^k_{p_{mf}}$ μ=0.093 σ^2=0.0014 κ=3.47 (c) $d^{corr}_{p_{mf}}$ μ=0.108 σ^2=0.0031 κ=2.52

Figure 5.12: Results of the bootstrap analysis. The subfigures show the histograms of the quartet distance values between the replicates taxonomies and the reference taxonomies, μ being the mean and σ^2 the variance and κ the kurtosis. The taxonomies are inferred with average link agglomerative clustering and with d_{corr} (a), $d^k_{p_{mf}}$ (b) and $d^{corr}_{p_{mf}}$ (c) distance.

archical model justified? To find the answers, we assessed the behavior of 11 different distance functions on the data. The three most promising measures were then applied to taxonomic hierarchy inference with agglomerative clustering and Fitch-Margoliash least squares method. To conclude how the data was structured, qualitative assessment was used. This was done by plotting the geographical co-occurrence of species at different levels of the inferred taxonomies. The suitability of a hierarchical model was assessed with quantitative methods, that is the Monte Carlo and that Bootstrap analysis. Our conclusions are:

1) The study suggested that the structure found in EMMA-data is related to geographical occurrence of the species. On a higher level of hierarchy, the inferred taxonomies divide the species roughly to a set of northern and southern species. At a lower level, more specific regions start to stand out: these being Fennoscandia, the Baltic region, central continental Europe, the Iberian Peninsula, the Alps and the eastern Mediterranean region. In general, mountain areas seem to be strongly present in the division of species at lower levels of the inferred taxonomies. Also, the analysis of variance on the climate profiles of that regions showed that the environmental conditions are significantly different in these regions. This gave further support for the validity of the geographical division.

2) The quantitative tests suggested that a hierarchical model for the co-occurrence of the species can be regarded reasonably justified when measuring similarity of occurrence with correlation distance d_{corr}. Other two applied measures, the probe distances $d^k_{p_{mf}}$ and $d^{corr}_{p_{mf}}$ derived from the sec-

ond Kulczynski distance and the correlation distance, gave less consistent support for a hierarchical model. The Monte Carlo analysis showed that the Goodman-Kruskal γ statistic yielded d_{corr} a clearly better fit to a hierarchy then what a non-hierarchical model would suggest. The fit of $d^{corr}_{p_{mf}}$ showed no significant difference to a non-hierarchical model, where as $d^{k}_{p_{mf}}$ showed even a significantly worse fit. Also, bootstrap analysis showed that, although the overall variance of taxonomic stucture under resampling is the smallest in the distribution of $d^{k}_{p_{mf}}$, the replicate hierarchies of d_{corr} result to a more compact tree distribution than with $d^{k}_{p_{mf}}$ or $d^{corr}_{p_{mf}}$.

Further discussion

We concluded in our study that an ultrametric hierarchical model is indeed supported when measuring the co-occurrence of species with d_{corr}. The test results failed to give similar consistent evidence to the considered probe distances $d^{k}_{p_{mf}}$ and $d^{corr}_{p_{mf}}$. The reason why the probe distances might have failed is that, no ecological structure was found in EMMA-data. Yet, our motivation behind the usage of probe measures was exactly the assumption of the existence of species niches related to ecology. The case most likely is that the 40×40 km^2 resolution of the data is not detailed enough to show ecological similarities in the species. Ecological niches found in ecosystem are probably more local and connected to local land types.

Another issue also concerning the probe distance came into our minds when analyzing the histograms of the different probe distances: most of the probe distances produced a normal-like distance value distribution on the EMMA-data. However, for distances behaving normally an ultrametric model will probably have a poor fit. Consider, for instance, randomly distributing a set of points inside a unit cube in Euclidean space. As the size of the set grows the points will distribute evenly inside the unit cube and the distance value distribution between the points will converge quite fast to a normal distribution. Hence, at least in this case, normality means a poor fit to an ultrametric model. Our guess is that, because the probe distance is in fact the sum of distances between the probes, it's behavior is affected by the *central limit theorem*. The sum of variables having a finite variance will be, according to the central limit theorem, approximately normally distributed [48, p. 156]. However, this hypothesis needs more study and will remain an interesting topic for further research.

Bibliography

[1] R. Agarwala, V. Bafna, M. Farach, M. Paterson, and M. Thorup. On the approximability of numerical taxonomy (fitting distances by tree metrics). *SIAM Journal on Computation*, 28(3):1073–1085, 1999.

[2] N. Ailon and M. Charikar. Fitting tree metrics: Hierarchical clustering and phylogeny. In *46th Annual IEEE Symposium on Foundations of Computer Science*, 2005. (to appear).

[3] N. L. Biggs. *Discrete Mathematics*. Oxford University Press, second edition, 2002.

[4] G. S. Brodal, R. Fagerberg, and C. N. S. Pedersen. Computing the quartet distance between evolutionary trees in time $O(n \log n)$. *Algorithmica*, 38:377–395, 2004.

[5] D. Bryant. *Building trees, hunting for trees, and comparing trees*. PhD thesis, University of Canterbury, 1997.

[6] A. H. Cheetham and J. E. Hazel. Binary (presence-absence) similarity coefficients. *Journal of Paleontology*, 43(5):1130–1136, September 1969.

[7] G. Das, H. Mannila, and P. Ronkainen. Similarity of attributes by external probes. In *Knowledge Discovery and Data Mining*, pages 23–29, 1998.

[8] A. C. Davison and D. V. Hinkley. *Bootstrap Methods and their Application*. Gambridge university press, 1997.

[9] L. R. Dice. Measurements of the amount of ecologic association between species. *Ecology*, 26:297–302, 1945.

[10] B. Efron. Bootstrap methods: another look at the jackknife. *The Annals of Statistics*, 7(1):1–26, 1977.

[11] B. Efron, E. Halloran, and S. Holmes. Bootstrap confidence levels for phylogenetic trees. In *proceedings of the National Academy of Sciences USA*, volume 93, pages 13429–13434, November 1996.

[12] G. F. Estabrook, F. R. McMorris, and C. A. Meacham. Comparison of undirected phylogenetic trees based on subtrees of four evolutionary units. *Systematic Zoology*, 34(2):193–200, 1985.

[13] B. S. Everitt. *Cluster Analysis*. Arnold Publishers, third edition, 1993.

[14] B. S. Everitt and G. Dunn. *Applied Multivariate Data Analysis*. Arnold Publisher, 1991.

[15] J. S. Farris. On the copenetic correlation coefficient. *Systematic Zoology*, 18:279–285, 1969.

[16] J. S. Farris. A successive approximations approach to character weighting. *Systematic Zoology*, 18:374–385, 1969.

[17] J. Felsenstein. PHYLIP - phylogeny inference package (version 3.2). *Cladistics*, 5:164–166, 1989.

[18] J. Felsenstein. *Inferring Phylogenies*. Sinuar Associates, Inc., 2004.

[19] W. M. Fitch and E. Margoliash. Construction of phylogenetic trees. *Science*, 155:279–284, January 1967.

[20] L. A. Goodman and W. H. Kruskal. Measure of association for cross classifications. *American Statistical Association Journal*, pages 732–764, December 1954.

[21] N. J. Gotelli and G. R. Graves. *Null Models in Ecology*. Smithsonian Institution Press, 1996.

[22] J. C. Gower and P. Legendre. Metric and euclidean properties of dissimilarity coefficients. *Journal of Classification*, 3:5–48, 1986.

[23] D. Gusfield. *Algorithms on Strings, Trees, and Sequences: Computer Science and Computational Biology*. Cambridge University Press, 1997.

[24] R. Hamming. Error detecting and error correcting codes. *The Bell System Technical Journal*, 26(2):147–161, April 1950.

[25] D. Hand, H. Mannila, and P. Smyth. *Principles of Data Mining*. MIT Press, 2001.

[26] C. Hennig and B. Hausdorf. Distance-based parametric bootstrap tests for clustering of species ranges. *Computational Statistics & Data Analysis*, 45(4):875–895, May 2004.

[27] R. Hijmans. WORLDCLIM global climate data set version 1.3. biogeo.berkeley.edu/worldclim/worldclim.htm.

[28] L. Hubert. Approximate evaluation techniques for the single-link and complete-link hierarchical clustering procedures. *Journal of the American Statistical Association*, 69(347):698–704, September 1974.

[29] P. Jaccard. Nouvelles recherches sur la distribution florale. *Bulletin de la Société Vaudoise des Sciences Naturelles*, 44:223–270, 1908.

[30] A. K. Jain and R. C. Dubes. *Algorithms for Clustering Data*. Prentice Hall, 1988.

[31] A. K. Jain, M. N. Murty, and P. J. Flynn. Data clustering: a review. *ACM Computing Surveys*, 31(3):264–323, 1999.

[32] J. Kim and T. Warnow. Tutorial on phylogenetic tree estimation. In *Intelligent Systems for Molecular Biology, Heidelberg*, 1999.

[33] S. Kullbach and R. A. Leibler. On information theory and sufficiency. *Annals of Mathematical Statistics*, 22:79–86, 1951.

[34] B. F. J. Manly. *Randomization and Monte Carlo Methods in Biology*. Chapman and Hall, 1991.

[35] N. Metropolis. The beginning of the monte carlo method. *Los Alamos Science*, (15):125–130, 1987.

[36] N. Metropolis and S. Ulam. The monte calo method. *Journal of American Statistical Association*, 44(247):335–341, september 1949.

[37] J. A. Miller. Assessing progress in systematics with continuous jackknife function analysis. *Systematic Biology*, 51(1):56–65, 2003.

[38] J. S. Milton and J. C. Arnold. *Introduction to Probability and Statistics: Principles and Applications for Engineering and the Computing Sciences*. McGraw-Hill international editions, 1990.

[39] T. Mitchell-Jones. Societas europaea mammalogica, 2005. www.european-mammals.org.

[40] P. Moen. *Attribute, Event Sequence, and Event Type Similarity Notations for Data Mining.* PhD thesis, Unversity of Helsinki, 2000.

[41] D. Penny, L. R. Foulds, and M. D. Hendy. Testing the theory of evolution by comparing phylogenetic trees constructed from five different protein sequences. *Nature*, 297:197–200, May 1982.

[42] J. B. Phipps. Dendrogram topology. *Systematic Zoology*, 20:306–308, 1971.

[43] A. C. Rencher. *Methods of Multivariate Analysis.* John Wiley & Sons, Inc., second edition, 2002.

[44] S. J. Russell and P. Norvig. *Artificial Intelligence: A Modern Approach.* Prentice Hall, second edition, 2003.

[45] J. Simpson and E. Weiner, editors. *Oxford English Dictionary.* OED Online Oxford University Press, second edition, 1989.

[46] M. A. Steel and D. Penny. Distributions of tree comparison metrics - some new results. *Systematic Biology*, 42(2):126–141, 1993.

[47] S. Theodoridis and K. Koutroumbas. *Pattern Recognition.* Academic Press, second edition, 2003.

[48] D. Williams. *Weighing the Odds.* Cambridge University Press, 2001.

[49] G. U. Yule. On the methods of measuring association between two attributes. *Journal of the Royal Statistical Society*, 75:579–653, May 1912.