Feature network models for proximity data : statistical inference, model selection, network representations and links with related models

Frank, L.E.

Chapter 3

Standard Errors, Prediction Error and Model Tests in Additive Trees [1]

Abstract

Theoretical standard errors and confidence intervals are given for the estimates of branch lengths in psychometric additive trees for a priori known tree topologies as well as for estimated tree topologies. A model test and an estimate of prediction error to compare different tree topologies are also given. The statistical inference theory proposed here differs from existing approaches due to the combination of the use of features with the multiple regression framework. Additive trees can be considered as a special case of Feature Network Models, where the objects are described by features, which are binary variables that indicate whether a particular characteristic is present or absent. Considering features as predictor variables leads in a natural way to the univariate multiple regression model.

3.1 Introduction

In general, there are two types of graphical representations of proximity data: spatial models and network models. The spatial models - such as multidimensional scaling - represent each object as a point in a coordinate space (usually Euclidean space) in such a way that the metric distances between the points approximate the observed proximities between the objects as closely as possible. In network models, the objects are represented as nodes in a connected graph, so that the spatial distances between the nodes in the graph approximate the observed proximities among the objects. In MDS, the primary objective is to find optimal coordinate values that lead to distances that approximate the observed proximities between the objects, whereas in network models, the primary objective is to find the correct set of *relations* between the objects that describe the observed proximities.

[1] This chapter has been submitted for publication as: Frank, L. E. & Heiser, W. J. (2005). Standard errors, prediction error and model tests in additive trees. *Submitted manuscript.* With an exception for the notes in this chapter, which are reactions to remarks made by the members of the promotion committee.

Feature Network Models or FNM (Heiser, 1998) represent proximity data in a discrete space, usually by a network representation. The relations between the objects are characterized by the kind of features they possess and by the combination of these features. Features are binary variables indicating for each object whether a particular characteristic is present or absent. The relations between the features, or the feature structure, determine the shape of the graphical representation, which is either a network or a tree. Therefore, FNM can be viewed as a general framework for graph representations, where the network is the general case and trees are special cases.

In FNM, the relation between any two objects *i* and *j* is represented by the symmetric set difference (= the difference between the union and the intersection of two sets) of the set of features that describes the two objects. The symmetric set difference expresses the number of features that object *i* possesses that are not shared by object *j* and vice versa, which amounts to the number of non-common elements of the objects. Applying the symmetric set difference on binary features in a binary coordinate space, corresponds to the *Hamming* distance, or the city-block distance. The relation between the objects can be expressed in terms of city-block distances, which is useful for graphical display purposes. Besides the graphical representation, the features in their own right are highly informative about the relations between the objects. In the final solution, each feature has a parameter value that indicates its relative importance: the *feature discriminability* value.

Since the introduction by Tversky (1977) of the Contrast Model, where objects are represented by subsets of discrete features, several different tree models have been developed in the psychological literature that are based on features (see Carroll & Corter, 1995, and Corter, 1996 for an overview). These models neither provide ways to estimate the standard errors of the parameter values, nor provide confidence intervals to assess the stability of the solution. In some psychometric applications (e.g. De Soete, 1983; Corter, 1996) least squares minimization is used to obtain the solution, treating the problem as a multiple regression model. Nevertheless, in the psychological literature, the statistical inference aspects of the multiple regression model have not been fully exploited for additive trees. The statistical inference theory proposed in this paper derives from the multiple regression framework because the use of features, when considered as predictor variables, leads in a natural way to the univariate multiple regression model. However, the standard multiple regression statistical inference theory cannot be applied because the network or additive tree representation imposes constraints on the model parameters. Negative edge lengths have no meaning in a network or an additive tree. In the context of FNM the implication is that the feature discriminability parameters associated with the features (the predictor variables) are constrained to be positive. These positivity constraints are even more relevant for additive tree representations because each branch in the tree is represented by a single feature, as will become clearer in this paper.

In contrast to the psychological tree domain, the phylogenetic tree domain does have a strong tradition of statistical inference. Important contributions in the field of statistical inference in phylogenies were made by Felsenstein (1985 and, for an overview, 2004, Chapters 19 - 21) and by Nei, Stephens, and Saitou (1985). Felsenstein (1983) evaluated the stability of a tree topology using the bootstrap to calculate the proportion of bootstrap trees that agree with the original tree in terms of topology and not directly in terms of branch lengths. In addition, the phylogenetic literature offers many examples of the estimation of the standard errors of branch lengths. The branch lengths are usually estimated with ordinary least squares, and the variances of the branch lengths are calculated by taking into account the method used to compute the evolutionary distances (Li, 1989; Nei et al., 1985; Rzhetsky & Nei, 1992; Tajima, 1992). Bulmer (1991) estimated the branch lengths and their standard errors with generalized least squares, which allows for correcting the correlation of distances between species that share one or more common paths. Despite the abundance of methods to compute standard errors for the branches of the phylogenetic trees, none of these methods take into account that when estimating the standard errors of the branch length estimates, one should correct for the fact that the estimates of the branch lengths have been constrained to be positive. The problem of biased estimates of the branch lengths has been diagnosed by Gascuel and Levy (1996), who correctly remark that the right way to estimate the edge lengths in phylogenies is to use linear regression under positivity constraints, and by Ota, Waddell, Hasegawa, Shimodaira, and Kishino (2000), who use a mixture of χ^2 distributions to construct appropriate likelihood ratio tests for nested evolutionary tree models. The mixture of χ^2 distributions is based on earlier results obtained by Self and Liang (1987) and Stram and Lee (1994) who derived limiting distributions of the likelihood ratio statistic when varying numbers of parameters are on the boundary. However, in the additive tree framework, Ota et al. (2000) have not made adjustments for the estimation of the standard errors of the branch lengths.

Recently, Frank & Heiser (in press *a*) showed how to compute standard errors and confidence intervals for the inequality constrained feature discriminability parameters in FNM. In this paper, we will show that the same statistical inference theory that has been proven to be useful for networks also applies to the family of tree representations. We propose a way to compute standard errors and confidence intervals for branch lengths of additive trees, and especially for tree topologies that include star shaped components, which means that one or more branches have edge lengths equal to zero (resulting from the correction of negative values). The multiple regression framework can be used to impose inequality constraints on the parameters and at the same time to compute theoretical standard errors for the inequality constrained least squares parameters that represent the edge lengths of the branches in an additive tree. These standard errors were introduced by Liew (1976) and take into account the fact that the parameter estimates are bounded below by zero. Whereas the results presented by Frank & Heiser (in press *a*) were limited to the situation of an a priori known feature structure (or tree topology), the present study shows that the same theory can be applied for the situation where the tree topology is not known in advance if the sample can be divided in a test set and a training set. Resulting from the same inequality constrained least squares framework, the paper shows an application of the Kuhn-Tucker test that is used to test whether the constrained solution is in accordance with the data. In addition, an easy way to estimate the prediction error of the model is provided, which allows for comparison of different tree topologies.

The remainder of this paper is organized as follows. It starts with a description of the Feature Network Models with an application on sample data, followed by an explanation of additive trees as special cases of FNM. Next, the statistical inference theory for inequality constrained least squares is introduced and evaluated with Monte Carlo simulation techniques. The first simulation study shows how to obtain the empirical p-value for the Kuhn-Tucker test. The second simulation study assesses the performance of the theoretical standard errors in comparison to bootstrap standard errors for the case where the tree topology is known in advance. It will become clear that the theoretical standard errors are much closer to the true values than the bootstrap standard errors and that the confidence intervals based on theoretical standard errors have better coverage performance than the bootstrap confidence intervals. The third simulation study shows that the same statistical inference theory can be applied in the situations where the tree topology is not known in advance and estimated with the neighbor-joining (NJ) method (Saitou & Nei, 1987). The NJ method is a widely used tree finding algorithm, especially in the phylogenetic domain, that is related to the ADDTREE algorithm by Sattath and Tversky (1977), which was developed in the mathematical psychology domain. Saitou and Nei (1987) and Gascuel (1994) have demonstrated that the NJ and the ADDTREE algorithms are strongly related and usually provide identical or very similar trees. A comparison between the statistical inference theory proposed for FNM in this paper and the statistical inference practice in the phylogenetic tree domain is provided in the discussion.

3.2 Feature Network Models

Since the general framework of this paper is the network representation, this section starts with a description of the Feature Network Models. FNM represent proximity data in a discrete space usually by a network representation. The properties of the models will be illustrated using a data set, the *kinship* data of Rosenberg and Kim (1975). A number of 165 female students and 165 male students were asked to group fifteen kinship terms on the basis of their similarities in minimally two and maximally fifteen categories. Half of the students were allowed to do the sorting task more than one time. Dissimilarity measures were derived for each pair of kinship terms by counting the number of subjects who placed the two terms in different categories. The data that were used in this study are the dissimilarity values of the female students ($n = 165$). Analyzing the dissimilarity matrix for the female students with the cluster differences scaling algorithm[2] of FNM (Heiser, 1998) yielded a solution with 5 features, displayed in Table 3.1. The features represent criteria most likely used by the female students to categorize the kinship terms.

Features are binary variables indicating for each object whether a particular characteristic is present or absent. Some set theoretic properties of the binary feature matrix lead to the estimation of a distance measure that approximates the observed dissimilarities. The difference between the union and intersection (= the symmetric set

[2] The first application of FNM used a cluster differences scaling algorithm (Heiser, 1998) with number of clusters equal to two, which constitutes a one-dimensional MDS problem with the coordinates restricted to form a bipartition. Because it is still a hard combinatorial problem, the implementation uses a nesting of several random starts together with K-means type of reallocations.

Table 3.1: The 5 binary features describing the kinship terms

Kinship terms	Gender	Nuclear family	Collaterals	Generation(1, 2)	Parent/child
aunt	0	0	1	1	0
brother	1	1	1	0	0
cousin	1	0	0	0	0
daughter	0	1	1	0	1
father	1	1	1	1	1
granddaughter	0	1	0	1	1
grandfather	1	1	0	1	0
grandmother	0	1	0	1	0
grandson	1	1	0	1	1
mother	0	1	1	1	1
nephew	1	0	0	0	1
niece	0	0	0	0	1
sister	0	1	1	0	0
son	1	1	1	0	1
uncle	1	0	1	1	0

difference) expresses the number of non-common features possessed by the objects i and j. For example, the symmetric set difference for the two kinship terms *aunt* and *cousin* is the set $\{Gender, Collaterals, Generation\}$. Following Goodman (1951, 1977) and Restle (1959, 1961), a distance measure that satisfies the metric axioms can be expressed as a simple count μ of the elements of the symmetric set difference between the stimuli O_i and O_j and becomes the *feature distance*: $d(O_i, O_j) = \mu[(O_i \cup O_j) - (O_i \cap O_j)]$.

If $\mathbf{E}$ is a binary matrix of order $m \times T$ that indicates which features t describe the m objects, as in Table 3.1, the re-expression of the feature distance in terms of coordinates is as follows (Heiser, 1998):

$$\begin{aligned} d(O_i, O_j) &= \mu[(O_i \cup O_j) - (O_i \cap O_j)] \\ &= \sum_t |e_{it} - e_{jt}|, \end{aligned} \tag{3.1}$$

where $e_{it} = 1$ if feature t applies to object i, and $e_{it} = 0$ otherwise. This re-expression of the feature distance in terms of binary coordinates is also known as the *Hamming* distance. The feature distance used in FNM is a weighted version of the distance in Equation 3.1:

$$d(O_i, O_j) = \sum_t \eta_t |e_{it} - e_{jt}|, \tag{3.2}$$

where the weights η_t express the relative contribution of each feature. Each feature splits the objects into two classes, and η_t measures how far these classes are apart.

Table 3.2: Feature parameters ($\hat{\eta}$), standard errors and 95% t-confidence intervals for Feature Network Model on *kinship* data with $R^2 = .95$.

Features	$\hat{\eta}$	$\hat{\sigma}_\eta$	95% CI	
Gender	27.54	0.63	26.31	28.77
Nuclear family	25.22	0.66	23.93	26.51
Collaterals	21.71	0.64	20.46	22.96
Generation (1,2)	18.58	0.64	17.33	19.83
Parent/child	15.06	0.64	13.81	16.31

For this reason, Heiser (1998) called the feature weight a *discriminability parameter*. The feature discriminability parameters are estimated by minimizing the following least squares loss function:

$$\min_{\hat{\boldsymbol{\eta}}} = \|\mathbf{X}\hat{\boldsymbol{\eta}} - \boldsymbol{\delta}\|^2, \tag{3.3}$$

where $\mathbf{X}$ is of size $n \times T$ and $\boldsymbol{\delta}$ is a $n \times 1$ vector of dissimilarities, with n equal to all possible pairs of m objects: $m(m-1)/2$. The problem in Equation 3.3 is expressed in a more convenient multiple linear regression problem, where the matrix $\mathbf{X}$ is obtained by applying the following transformation on the rows of matrix $\mathbf{E}$ for each pair of objects, where the elements of $\mathbf{X}$ are defined by:

$$e_{lt} = |e_{it} - e_{jt}|, \tag{3.4}$$

where the index $l = 1, \cdots, n$ varies over all pairs (i, j). The result is the binary $(0, 1)$ matrix $\mathbf{X}$, where each row ($\mathbf{x}'$) represents the distinctive features for some pair of objects, with 1 meaning that the feature is distinctive for a pair of objects. The weighted sum of these distinctive features is the fitted distance for each pair of objects and is equal to $\mathbf{d} = \mathbf{X}\boldsymbol{\eta}$. Corter (1996, Appendix C, p. 57) uses a similar matrix $\mathbf{X}$ in the linear regression context to obtain the lengths of the branches in an additive tree.

Table 3.2 shows the feature discriminability parameters $\hat{\eta}_t$ obtained by PROXGRAPH, the program developed in Matlab to fit the FNM. The five features solution explains 95.35% of the variance in the data, and the values of the feature parameters lead to the conclusion that the most important categorizing criteria were: *Gender*, *Nuclear family*, and *Collaterals*. All five features played a more or less important role in categorizing the kinship terms as follows from the 95% t-confidence intervals that show that all feature parameters differ significantly from zero (Table 3.2).

Figure 3.1 shows the Feature Network representation that results from the fitted distances on the *kinship* data. The kinship terms are the vertices in the network and the feature distances ($\hat{\mathbf{d}} = \mathbf{X}\hat{\boldsymbol{\eta}}$) are represented as the sum of the edge lengths along the shortest path in the graph, where the edge lengths are the feature parameters $\hat{\boldsymbol{\eta}}$. How the network is obtained will be explained in the following section. The five-dimensional feature network has been embedded in 3-dimensional Euclidean space using PROXSCAL[3], a multidimensional scaling program distributed as part of

[3] with the interval transformation option and initialized with the simplex solution

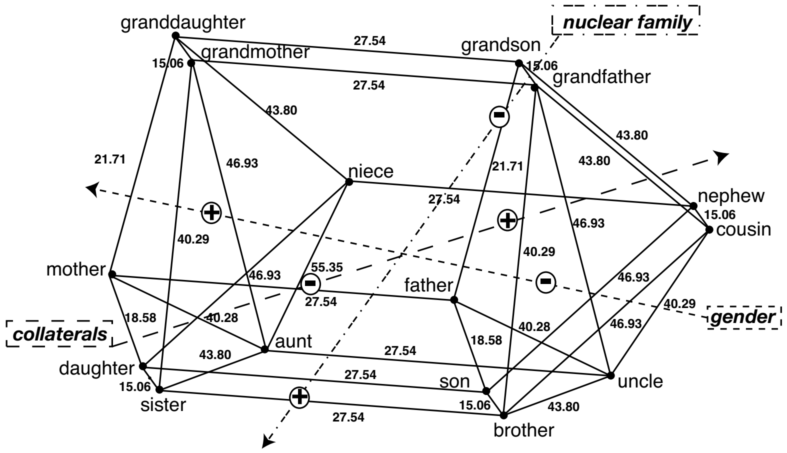

Figure 3.1: Feature Network representation for the *kinship* data with the three most important features (*Gender*, *Nuclear family* and, *Collaterals*) represented as vectors. The plus and minus signs designate the projection onto the vector of the centroids of the objects that posses the feature (+) and the objects that do not have that feature (-).

the Categories package by SPSS (Meulman & Heiser, 1999). The solution of the common space was restricted by a linear combination of the feature variables that are represented as vectors in Figure 3.1, leading from the origin through the point with coordinates equal to the correlations of each feature with each of the three dimensions. The network clearly shows the distinction between the female kinship terms and the male kinship terms produced by the most important feature *Gender*. This feature as well as the second and third most important features *Nuclear family* and *Collaterals* are represented by vectors in the network. The plus and minus signs on each vector designate the projection onto the vector of the centroids of the kinship terms that posses the feature $(+)$ and the kinship terms that do not possess that feature $(-)$.

3.3 Feature Network Models: network and additive tree representations

The relations between the features in FNM determine the shape of the network. A set of overlapping features will result in a network graph, which is a connected graph with cycles. When the set of features has a nested structure, i.e., all pairs of features are either nested or disjoint, the network will have the shape of an unrooted additive tree, a graph without cycles (Buneman, 1971). If the unrooted additive tree is a bifurcating tree, there are fixed numbers of edges (branches), internal and external nodes, given a number of objects m (*cf.* Felsenstein, 2004, Chapter 3). Bifurcating trees have interior nodes of degree 3, meaning that each internal node connects to

	Features								
	F1	F2	F3	F4	F5	F6	F7	F8	F9
Objects									
a	1	1	0	1	0	0	0	0	0
b	1	1	0	0	1	0	0	0	0
c	1	0	1	0	0	1	0	0	0
d	1	0	1	0	0	0	1	0	0
e	0	0	0	0	0	0	0	1	0
f	0	0	0	0	0	0	0	0	1
Internal nodes									
(e,f)	0	0	0	0	0	0	0	0	0
(a,b,c,d)	1	0	0	0	0	0	0	0	0
(a,b)	1	1	0	0	0	0	0	0	0
(c,d)	1	0	1	0	0	0	0	0	0

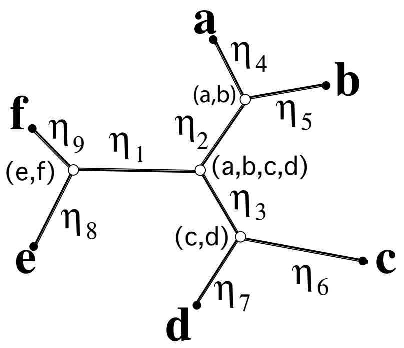

Figure 3.2: Nested and disjoint feature structure and corresponding additive tree representation. Each edge in the tree is represented by a feature and the associated feature discriminability parameter η_t.

three other nodes (internal or external) and every external node (or leaf node) is of degree 1, which means that only one branch leads to an external node. Given these specifications, the bifurcating unrooted additive tree for m objects has a number of $2m - 3$ edges because for each new object added to an existing tree an internal node and two new edges must be added (*cf.* Felsenstein, 2004, Chapter 3). Following this reasoning, the number of internal nodes is fixed to $(2m - 3 - 1)/2$. In contrast to the bifurcating trees, the multifurcating trees do not have a fixed number of edges and nodes for a given number of objects. Since the degree of each internal node in multifurcating trees is not necessarily equal to 3, there exists a range of possible numbers of internal nodes and numbers of edges that depend on the number of internal nodes.

In terms of features, the bifurcating unrooted additive tree has a set of $T = 2m - 3$ nested features and the internal nodes are represented by $(T - 1)/2$ supplementary objects added to the original set of objects in the feature matrix. Figure 3.2 shows the feature matrix and the corresponding tree graph for an example of 6 objects. There are $T = 2m - 3 = 9$ nested features, $m = 6$ leaf nodes and, $n_o = (T - 1)/2 = 4$ internal nodes. The nested structure of the features becomes apparent: the features either exclude each other or one is a subset of the other. Each cluster in the tree, for example the bipartition of the objects a and b against the other objects, is represented by a *cluster feature*, that is, a feature which describes more than one object, in this example feature F_2. The internal nodes are defined as supplementary objects with a feature pattern that is the intersection of the feature patterns of a subset of the objects. Therefore, they can be labeled by listing the objects in the subset. The leaf nodes of the tree represent the 6 objects and the associated edges correspond to *unique features*, which are features that belong to one object exclusively. In the example in Figure 3.2

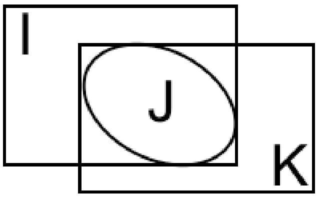

Figure 3.3: Betweenness holds when $J = I \cap K$, where I, J, and K are sets of features describing the corresponding objects i, j, and k.

the unique features are the set $\{F_4, F_5, F_6, F_7, F_8, F_9\}$. Note that these unique features are also either nested or disjoint with respect to the cluster features.

Additive tree representation and feature distance

The feature distance parallels the path-length distance in a valued graph when one of the metric axioms, the triangle inequality, is reaching its limiting additive form $d_{ik} = d_{ij} + d_{jk}$ (Flament, 1963; Goodman, 1951, 1977; Heiser, 1998). In a network graph, each time that the distance d_{ik} is exactly equal to the sum $d_{ij} + d_{jk}$ the edge between the objects i and k can be excluded, resulting in a parsimonious subgraph of the complete graph.

In terms of features the condition $d_{ik} = d_{ij} + d_{jk}$ is reached when object j is *between* objects i and k. The objects can be viewed as sets of features: $\mathcal{S}_i$, $\mathcal{S}_j$, and $\mathcal{S}_k$. Betweenness of $\mathcal{S}_j$ depends on the following conditions (Restle, 1959):

1. $\mathcal{S}_i$ and $\mathcal{S}_k$ have no common members which are not also in $\mathcal{S}_j$;

2. $\mathcal{S}_j$ has no unique members which are in neither $\mathcal{S}_i$ nor $\mathcal{S}_k$.

Figure 3.3 clearly shows that betweenness holds when the set $\mathcal{S}_j$ is exactly equal to the intersection of the sets $\mathcal{S}_i$ and $\mathcal{S}_k$ - in that case $\mathcal{S}_j$ has no unique features (Tversky & Gati, 1982) -, or when the set $\mathcal{S}_j$ consists of a subset of the intersection of the sets $\mathcal{S}_i$ and $\mathcal{S}_k$. In both situations $d_{ik} = d_{ij} + d_{jk}$. In the following, it will become clear that an additive tree structure results from a special feature structure where there always is an internal node $\mathcal{S}_j$ between any two leaf nodes $\mathcal{S}_i$ and $\mathcal{S}_k$.

An additive tree is a special subgraph of the complete graph, where each edge is represented by a separate feature. The edges leading directly to leaf nodes correspond to unique features, the set of features that describe only one object (see Figure 3.2). A nested set of features is not sufficient to produce a tree graph with FNM. A set of internal nodes has to be added to the set of objects (the external nodes). These internal nodes play the role of the set $\mathcal{S}_j$ in the betweenness condition by forcing the betweenness to hold exactly for any pair of objects i and k that have an associated nested set of features, leaving only paths between objects that are in an hierarchical relation to each other. Each edge between two internal nodes corresponds exactly to one cluster feature, and the edge length to its weight (see Figure 3.2).

It should be noted that the estimated distances between the internal nodes in the tree cannot be compared to dissimilarities because these quantities are not observed. To calculate all distances simultaneously requires a modification of the original feature matrix **E** (Equation 3.1). The feature matrix **E** is augmented with a supplementary set of objects equal to the number of internal nodes.

The augmented $\mathbf{E}_T$ matrix is as follows:

$$\mathbf{E}_T = \begin{bmatrix} \mathbf{E}_C & \mathbf{E}_U \\ \mathbf{E}_N & \mathbf{E}_0 \end{bmatrix}, \tag{3.5}$$

where $\mathbf{E}_C$ is a $m \times T_{\mathrm{C}}$ matrix, representing the set of cluster features and $\mathbf{E}_U$ is a $m \times T_{\mathrm{U}}$ matrix representing the set of unique features. Both parts describe the set of observed objects. The remaining two parts are related to the set of internal nodes (n_o): $\mathbf{E}_N$ is of size $n_o \times T_{\mathrm{C}}$ and $\mathbf{E}_0$ contains zeros only and has size $n_o \times T_{\mathrm{U}}$. Each row of $\mathbf{E}_N$ and $\mathbf{E}_0$ represents the feature pattern of each node. This nodal feature pattern is equal to the intersection of the feature patterns belonging to the objects (the rows of $\mathbf{E}_C$ and $\mathbf{E}_U$) that are represented by each particular node. The intersection of the feature patterns related to the unique features is always zero and, consequently, $\mathbf{E}_0$ contains zeros only. Figure 3.2 shows the four parts of the augmented $\mathbf{E}_T$ matrix. The objects (a, b, c, d, e, f) are described with cluster features and with unique features: the part with the cluster features, $\mathbf{E}_C$, is formed by the set features $\{F_1, F_2, F_3\}$, the part of the unique features, $\mathbf{E}_U$, is formed by $\{F_4, F_5, F_6, F_7, F_8, F_9\}$. The feature patterns of the internal nodes are represented by the parts $\mathbf{E}_N$ and $\mathbf{E}_0$. The $\mathbf{E}_0$ part is related to the unique features and contains zero's only. The $\mathbf{E}_N$ relates to the cluster features and the feature pattern of each internal node is formed by taking the intersection of the feature pattern belonging to the corresponding objects. For example, the feature pattern for internal node (a, b) is formed by taking the intersection of the feature pattern for object $a = \{110\}$ and object $b = \{110\}$, resulting in the feature pattern $\{110\}$.

Dissimilarities are only available for the objects and not for the internal nodes. Therefore, the feature discriminability parameters $\boldsymbol{\eta}$ are estimated using only the parts $\mathbf{E}_C$ and $\mathbf{E}_U$. After applying the featurewise distance transformation in Equation 3.4 to the matrix $[\mathbf{E}_C \quad \mathbf{E}_U]$, the resulting matrix **X** is used to obtain the estimates of the feature discriminability parameters ($\hat{\boldsymbol{\eta}}$) by minimizing the loss function in Equation 3.3. To obtain the estimated distances for the edges that are linked to internal nodes, the featurewise distance transformation (Equation 3.4) is applied to the augmented matrix $\mathbf{E}_T$, yielding the matrix $\mathbf{X}_T$. The estimated feature distances for the complete tree are equal to $\hat{\mathbf{d}}_T = \mathbf{X}_T\hat{\boldsymbol{\eta}}$. Given this description, it is easy to understand that every tree topology, known by theory or resulting from any tree constructing algorithm, can be transformed into an augmented feature matrix $\mathbf{E}_T$, such that, when analyzed as FNM with PROXGRAPH, it will lead to a tree representation of the data.

Example of additive tree obtained with feature structure

An example of a multifurcating additive tree is the solution obtained by De Soete and Carroll (1996) on the *kinship* data. The augmented $\mathbf{E}_T$ based on this given tree

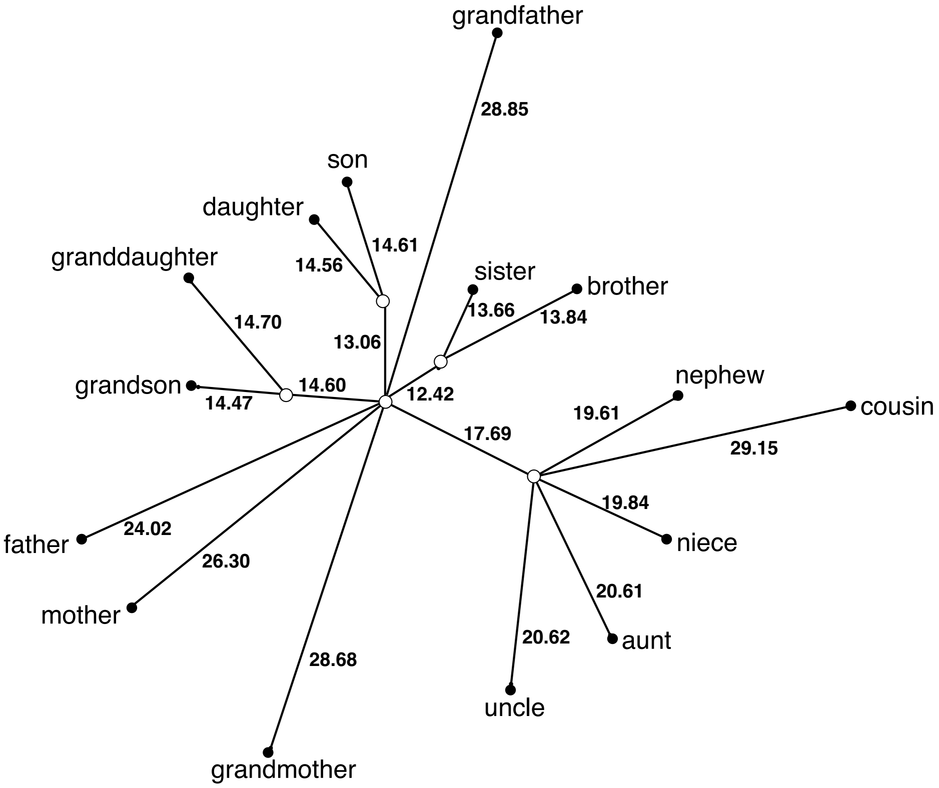

Figure 3.4: Unresolved additive tree representation of the *kinship* data based on the solution obtained by De Soete & Carroll (1996).

topology is displayed in the first part of Figure 3.5 and yields the additive tree representation in Figure 3.4. The 2-dimensional embedding of the tree has been obtained by submitting Euclidean distances calculated on the augmented $\mathbf{E}_T$ to the MDS program PROXSCAL[4], a multidimensional scaling program distributed as part of the Categories package by SPSS (Meulman & Heiser, 1999). The associated feature parameters and 95% t-confidence intervals are given in Figure 3.6. The construction of the confidence intervals will be explained in the next section. Some of the feature parameters have zero values ($F_2, F_3, F_4, F_{21}, F_{23}$) leading to the unresolved tree representation of Figure 3.4. The expected number of nodes is $12 + 1 = 13$ with 15 objects, but only 6 internal nodes remain in the final solution due to activation of the positivity constraints. The feature structure ($\mathbf{E}_T$) can therefore be simplified to the matrix shown in the second part of Figure 3.5.

[4] allowing a ratio scale transformation with a simplex start.

Feature structure resolved tree

	Nuclear family											Collaterals									Gandparents/Grandchildren						
	F1	F2	F3	F4	F5	F6	F7	F8	F9	F10	F11	F12	F13	F14	F15	F16	F17	F18	F19	F20	F21	F22	F23	F24	F25	F26	F27
aunt	0	0	0	0	0	0	0	0	0	0	0	1	1	1	0	1	0	0	0	0	0	0	0	0	0	0	0
brother	1	0	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
cousin	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0
daughter	1	1	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
father	1	1	0	0	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
granddaughter	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	1	0	0	0
grandfather	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1	0	1
grandmother	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1	1	0
grandson	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	0	0	0	0
mother	1	1	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
nephew	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0
niece	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0
sister	1	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
son	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
uncle	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0
nodes																											
1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
3	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
4	1	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
5	1	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
6	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
7	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
8	0	0	0	0	0	0	0	0	0	0	0	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0
9	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
10	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0
11	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0
12	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1	0	0
13	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Feature structure unresolved tree

	Nuclear family + Grandparents/Grandchildren																Collaterals					
	F1	F5	F6	F7	F8	F9	F10	F11	F12	F13	F14	F15	F16	F17	F18	F19	F20	F22	F24	F25	F26	F27
aunt	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0
brother	1	0	0	0	0	0	0	1	1	0	1	0	0	0	0	0	0	0	0	0	0	0
cousin	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
daughter	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
father	1	0	0	0	0	0	0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0
granddaughter	1	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
grandfather	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0
grandmother	1	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0
grandson	1	0	0	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
mother	1	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0
nephew	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1
niece	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0
sister	1	0	0	0	0	0	0	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0
son	1	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
uncle	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
nodes																						
4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
5	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
9	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
10	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
12	1	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
13	1	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0

Figure 3.5: Feature structure for the resolved additive tree representation (*top*) of the *kinship* data and simplified feature structure for the unresolved additive tree representation (*bottom*) of Figure 3.4.

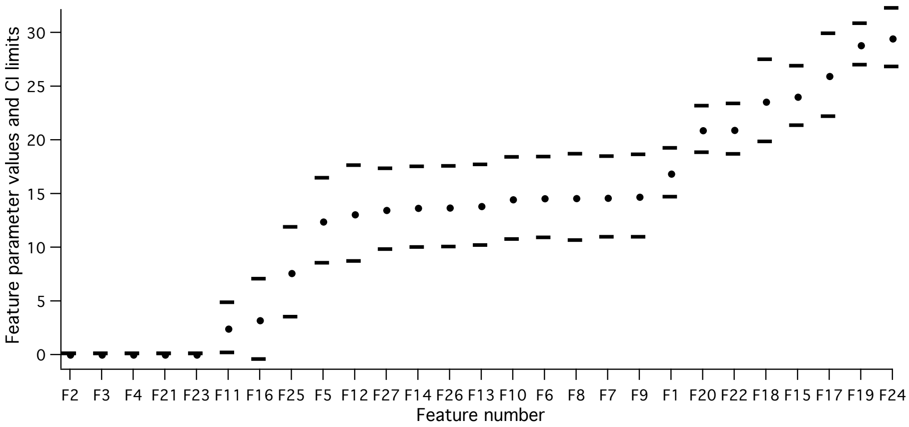

Figure 3.6: Feature parameters ($\hat{\eta}_{\mathrm{ICLS}}$) and 95% t-confidence intervals for additive tree solution on *kinship* data with $R^2 = .96$.

3.4 Statistical inference in additive trees

This section shows how the multiple linear regression framework can be used to obtain several statistical inference measures for additive trees. The features of an additive tree can be considered as predictor variables and the feature discriminability parameters are estimated like regression coefficients, with the major difference that positivity constraints are imposed on the feature discriminability parameters, because they represent edge lengths in the tree representation. This section shows how to obtain standard errors for the inequality constrained least squares estimators that can be used to construct 95% t-confidence intervals for the feature discriminability parameters. The statistical inference theory is intended for the case where the tree topology is known in advance, but can also be applied when the tree topology is unknown, as will be shown in the following. This section also provides an application of the Kuhn-Tucker test that is used to test whether the constrained solution is in accordance with the data and results from the same theory used to obtain the standard errors. The last topic of this section provides a way to estimate prediction error with the *generalized cross-validation (GCV)* statistic. This estimate of prediction error combines the the analytical approximation of leave-one-out cross-validation commonly used in linear fitting methods with the inequality constrained least squares theory.

Obtaining standard errors for additive trees

An important difference of the current approach compared to what is usually done in the phylogenetic domain is that phylogenetic trees do not use explanatory variables like the features. In the case that the feature structure is known, the distinctive-feature additivity allows for considering the additive tree as a univariate multiple linear regression model:

$$\boldsymbol{\delta} = \mathbf{X}\boldsymbol{\eta} + \boldsymbol{\epsilon} \tag{3.6}$$

where $\boldsymbol{\delta}$ is a $n \times 1$ vector with dissimilarities, $\mathbf{X}$ is a known $n \times T$ binary $(0, 1)$ matrix of rank T and $\boldsymbol{\eta}$ is a $T \times 1$ vector. Each row of the matrix $\mathbf{X}$ results from the operation $\mathbf{x}_l = |\mathbf{e}_{it} - \mathbf{e}_{jt}|$ (Equation 3.4). For an additive tree representation, $\mathbf{X}$ contains the featurewise distances that result from the matrix $\mathbf{E}_T$ formed by the set of cluster features and unique features, as explained in the previous section.

We assume, like Ramsay (1982), that $\boldsymbol{\epsilon}$ in Equation 3.6 is a $n \times 1$ random vector that follows a normal distribution with constant variance σ^2 over replications of judgments,

$$\boldsymbol{\epsilon} \sim \mathcal{N}(\mu, \sigma^2\mathbf{I}), \tag{3.7}$$

where $\mathbf{I}$ is an identity matrix of rank n, and where it is assumed that σ^2 is small enough to ensure the occurrence of negative dissimilarities to be negligible. The parameters of the vector $\boldsymbol{\eta}$ are subject to positivity constraints because they represent edge lengths of the tree. As explained in the beginning of this section, the phylogenetic domain does not apply positivity constraints when estimating branch lengths and trees that yield negative branch length estimates are simply discarded. Hence, the phylogenetic domain might benefit from the following theory on inequality constrained least squares estimation.

The inequality constrained least squares estimator $\hat{\eta}_{\text{ICLS}}$ results from the quadratic programming problem (*cf.* Björk, 1996):

$$\begin{aligned} \min_{\boldsymbol{\eta}} &= (\boldsymbol{\delta} - \mathbf{X}\boldsymbol{\eta})'(\boldsymbol{\delta} - \mathbf{X}\boldsymbol{\eta}) \\ &\text{subject to } \mathbf{A}\boldsymbol{\eta} \geq \mathbf{r}, \end{aligned} \tag{3.8}$$

where the matrix of constraints $\mathbf{A}$ is a $C \times T$ matrix of rank C, and $\mathbf{r}$ is a $C \times 1$ null-vector because all parameters are constrained to be greater than or equal to zero.

The duality theory of the quadratic programming problem of Equation 3.8 is the basis for the estimation of the standard errors of the parameters (Liew, 1976) and results in the following expression of the estimator $\hat{\eta}_{\text{ICLS}}$ in terms of the dual solution:

$$\hat{\boldsymbol{\eta}}_{\text{ICLS}} = (\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\boldsymbol{\delta} + (\mathbf{X}'\mathbf{X})^{-1}\mathbf{A}'\tfrac{1}{2}\boldsymbol{\lambda}_{\text{KT}}, \tag{3.9}$$

where $\boldsymbol{\lambda}_{\text{KT}}$ is the vector with Kuhn-Tucker multipliers that results from solving the quadratic programming problem with Algorithm AS 225 (Wollan & Dykstra, 1987). As shown by Liew (1976) the estimated standard errors for the ICLS estimator vector are

$$\hat{\sigma}_{\text{ICLS}} = \sqrt{\hat{\sigma}^2 \text{diag}\left[\mathbf{M}(\mathbf{X}'\mathbf{X})^{-1}\mathbf{M}'\right]}, \tag{3.10}$$

where

$$\hat{\sigma}^2 = \left[(\boldsymbol{\delta} - \mathbf{X}\hat{\boldsymbol{\eta}}_{\text{OLS}})'(\boldsymbol{\delta} - \mathbf{X}\hat{\boldsymbol{\eta}}_{\text{OLS}})\right] / (n - T), \tag{3.11}$$

and

$$\mathbf{M} = \mathbf{I} + \text{diag}[(\mathbf{X}'\mathbf{X})^{-1}\mathbf{A}'\tfrac{1}{2}\boldsymbol{\lambda}_{\text{KT}}][\text{diag}(\hat{\boldsymbol{\eta}}_{\text{OLS}})]^{-1}. \tag{3.12}$$

If the model is unconstrained, the estimated variance-covariance matrix reduces to the variance-covariance matrix of the ordinary least squares (OLS) estimator. For more details the reader is referred to Liew (1976), Wolak (1987), and Frank & Heiser (in press *a*). The standard errors for the ICLS estimator can be used to construct 95% *t*-confidence intervals in the usual way.

When the tree topology is not known yet and has to be estimated from the sample data first, the theory described in the previous paragraph cannot be applied directly. The standard errors cannot be estimated on the same data that were used to obtain the tree topology. In practice, the problem can be circumvented by dividing the sample in a training set and a test set. The training set is used to derive the tree topology, which is fitted on the test data to obtain the standard errors and the 95% *t*-confidence intervals for the feature discriminability parameters. The rationale behind this approach is the following: assuming that the sample is an adequate representation of the population, the training set will yield a tree topology that is close to the population tree or feature set. The deviations from the true tree topology are assumed to result from sampling error, and, therefore, will probably lead to near zero feature discriminability values and confidence intervals that contain the value zero. These assumptions have been verified by Monte Carlo simulation and the results are provided in the following.

Testing the appropriateness of imposing constraints

In the previous it has been assumed that there exists a representation of the data in terms of (positive) distances between points in a network or a tree. The validity of this assumption can be verified in a hypothesis-testing framework: we can test whether the data is consistent with true values of the parameters satisfying the restrictions imposed on the estimated coefficients. The null hypothesis of the inequality constraints $\mathbf{A}\hat{\boldsymbol{\eta}}_{\text{ICLS}} \geq \mathbf{r}$ (the ICLS solution) can be tested against an unrestricted alternative $\hat{\boldsymbol{\eta}}_{\text{OLS}} \in \mathbf{A}^t$ (the OLS solution). These multivariate inequality constraints lead to the following likelihood ratio test:

$$-2ln\left(\frac{L_{\text{ICLS}}}{L_{\text{OLS}}}\right) = 2(lnL_{\text{OLS}} - lnL_{\text{ICLS}}), \tag{3.13}$$

where L_{ICLS} and L_{OLS} are the maximum values of the likelihood function under the null hypothesis $\mathbf{A}\boldsymbol{\eta} \geq \boldsymbol{r}$ and the alternative hypothesis $\boldsymbol{\eta} \in \mathbf{A}^t$, repectively. If σ^2 is known the LR statistic takes the following form:

$$LR = \left[(\boldsymbol{\delta} - \mathbf{X}\hat{\boldsymbol{\eta}}_{\text{ICLS}})'(\boldsymbol{\delta} - \mathbf{X}\hat{\boldsymbol{\eta}}_{\text{ICLS}}) - (\boldsymbol{\delta} - \mathbf{X}\hat{\boldsymbol{\eta}}_{\text{OLS}})'(\boldsymbol{\delta} - \mathbf{X}\hat{\boldsymbol{\eta}}_{\text{OLS}})\right]/\sigma^2. \tag{3.14}$$

According to Wolak (1987) the LR statistic is also the optimal value of the objective function, or the primal function of the following quadratic programming problem:

$$\min_{\boldsymbol{\eta}} = \left[(\boldsymbol{\delta} - \mathbf{X}\boldsymbol{\eta})'(\boldsymbol{\delta} - \mathbf{X}\boldsymbol{\eta}) - (\boldsymbol{\delta} - \mathbf{X}\hat{\boldsymbol{\eta}}_{\text{OLS}})'(\boldsymbol{\delta} - \mathbf{X}\hat{\boldsymbol{\eta}}_{\text{OLS}})\right]/\sigma^2$$
$$\text{subject to } -\mathbf{A}\boldsymbol{\eta} \geq \mathbf{r}. \tag{3.15}$$

Wolak (1987) showed that the Kuhn-Tucker test statistic (KT) is equal to the LR test statistic using the theory of quadratic programming, which states that the optimal value of the objective function of the primal equals that same value for the dual problem under certain conditions. The necessary conditions are that $\mathbf{X}'\mathbf{X}$ is nonsingular and $\mathbf{A}(\mathbf{X}'\mathbf{X})^{-1}\mathbf{A}'$ is positive definite. The Kuhn-Tucker test statistic is the optimal value of the dual problem of the objective function of Equation 3.13, and can be formulated as follows:

$$KT = \left[\boldsymbol{\lambda}'_{\mathrm{KT}}\mathbf{A}(\mathbf{X}'\mathbf{X})^{-1}\mathbf{A}'\boldsymbol{\lambda}_{\mathrm{KT}}\right] / 4\sigma^2 \qquad (3.16)$$

Wolak (1987) also showed that the *KT* and the *LR* statistics have the same distributions and continue to possess the same distribution if the same estimate for σ^2 is used when σ^2 is unknown and replaced by its estimated value $\hat{\sigma}^2$. The null distribution of both test statistics is a weighted sum of Snedecor's F distributions, a property that also holds for covariance matrices other than $\sigma^2\mathbf{I}$. For the hypothesis testing problem $\mathbf{H}_0 : \boldsymbol{\lambda}_{\mathrm{KT}} = 0$ versus $\mathbf{H}_1 : \boldsymbol{\lambda}_{\mathrm{KT}} \geq 0$ (which is equivalent to the testing problem $\mathbf{H}_0 : \mathbf{A}\boldsymbol{\eta} \geq \mathbf{r}$ versus $\mathbf{H}_1 : \boldsymbol{\eta} \in \mathbf{A}^T$), the null distribution of the *KT* statistic (and the *LR* statistic) with σ^2 replaced by $\hat{\sigma}^2$ (Equation 3.11), is equal to:

$$\begin{aligned} Pr_{0,4\hat{\sigma}^2\boldsymbol{\Lambda}}\left[KT \geq q\right] &= \sum_{c=1}^{C} Pr[\mathrm{F}_{c,n-T} \geq \tfrac{q}{C}]w(C,c,4\boldsymbol{\Lambda}) \\ Pr_{0,4\sigma^2\boldsymbol{\Lambda}}\left[KT = 0\right] &= w(C,0,4\boldsymbol{\Lambda}), \end{aligned} \qquad (3.17)$$

where $\boldsymbol{\Lambda} = (\mathbf{A}(\mathbf{X}'\mathbf{X})^{-1}\mathbf{A}')^{-1}$, and q is the value of the Kuhn-Tucker test statistic. The weights w denote the proportion of times $\boldsymbol{\lambda}_{\mathrm{KT}}$ (Equation 3.16) has exactly c elements larger than zero and can be calculated in closed form for the cases in which $C \leq 4$ (see Wolak, 1987, Appendix). For the cases where the number of constraints exceeds the number 4, Monte Carlo techniques can be used, as will be explained in the Method section.

Estimating prediction error

In addition to the the Kuhn-Tucker test that requires one of the models to be nested within the other (a constrained model versus an unconstrained model), there is an easy way to evaluate the goodness-of-fit of models that are not necessarily nested within each other. Likelihood ratio tests are not suited for testing nonnested models, which have the same number of effective parmeters (Felsenstein, 2004, pp. 316-318; Huelsenbeck & Rannala, 1997). Therefore, Felsenstein (1985, 2004) evaluates the goodness of fit of the tree topology by constructing a consensus tree using a resampling strategy (the nonparametric bootstrap). The *AIC* statistic (Akaike, 1974) can be used for any pair of models whether nested or not, and has been used for that purpose in phylogenetics (Kishino & Hasegawa, 1990), but also in several MDS applications (Takane, 1981, 1983; Takane & Carroll, 1981; Winsberg & Ramsay, 1981) and is mainly suitable when a log-likelihood loss function is used. Here, we propose a criterion closely related to AIC that is frequently used in the context of linear models: the *generalized cross-validation* (*GCV*). This statistic provides a convenient approximation to leave-one-out cross-validation for linear fitting under squared-error loss (Hastie, Tibshirani, & Friedman, 2001, p. 216). A linear fitting method is one for which we can write:

$$\hat{\boldsymbol{y}} = \mathbf{S}\boldsymbol{y}. \tag{3.18}$$

The hat matrix **S** from Equation 3.18 is equal to the combination of matrices that transforms the observed data **y** into the predicted values $\hat{\mathbf{y}}$.

Using the hat matrix, linear fitting methods can be written as follows,

$$\frac{1}{N}\sum_{i=1}^{N}[y_i - \hat{y}_i]^2 = \frac{1}{N}\sum_{i=1}^{N}\left[\frac{y_i - \hat{y}_i}{1 - S_{ii}}\right]^2, \tag{3.19}$$

where S_{ii} is the ith diagonal element of **S**. The GCV approximation is

$$GCV = \frac{1}{N}\sum_{i=1}^{N}\left[\frac{y_i - \hat{f}(x_i)}{1 - trace(\mathbf{S})/N}\right]^2, \tag{3.20}$$

where the quantity $trace(\mathbf{S})$ is the effective number of parameters. Applied to the additive tree the *generalized cross-validation* statistic can be computed as follows. From Liew (1976) we know that the following relation exists between the ICLS and the OLS estimator, which leads to the matrices needed to construct the hat matrix:

$$\begin{aligned} \hat{\boldsymbol{\eta}}_{\text{ICLS}} &= \mathbf{M}\hat{\boldsymbol{\eta}}_{\text{OLS}} \\ &= \mathbf{M}(\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\boldsymbol{\delta}. \end{aligned} \tag{3.21}$$

From the relation expressed in Equation 3.21 it follows that the predicted distance values can be obtained with:

$$\hat{\mathbf{d}} = \mathbf{X}\mathbf{M}(\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\boldsymbol{\delta}, \tag{3.22}$$

and, consequently, the hat matrix is equal to

$$\mathbf{S} = \mathbf{X}\mathbf{M}(\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'. \tag{3.23}$$

The *generalized cross-validation* error for the additive tree can be estimated using the trace of the hat matrix from Equation 3.23:

$$GCV_{\text{FNM}} = \frac{1}{n}\sum_{l=1}^{n}\left[\frac{\delta_l - \hat{d}}{1 - trace(\mathbf{S})/n}\right]^2. \tag{3.24}$$

3.5 Method Monte Carlo simulations

To evaluate the performance of the statistical inference theory described in the previous section, three Monte Carlo simulations were conducted using data structures that approximate the practice of data analysis with additive tree models. The first simulation shows how to obtain the empirical p-value for the Kuhn-Tucker test described in Equations 3.16 and 3.17. The second simulation study evaluates the performance of the nominal standard errors for known tree topologies compared to empirical (bootstrap) standard errors. The third simulation study assesses the performance of the nominal standard errors when the tree topology is unknown. In this study the performance of the GCV_{FNM} statistic that serves as an approximation for the prediction error is evaluated as well. All simulation procedures were programmed in Matlab and made use of its pseudo-random number generator, which was set to 1.0 prior to the simulation process.

Empirical *p*-value Kuhn-Tucker test

The null distribution of the Kuhn-Tucker test was calculated by simulating many data sets from a fixed population distribution. The model parameters were estimated from the original data (the *kinship* data) under the null hypothesis, i.e. the inequality constrained least squares model with associated feature parameters as displayed in Table 3.2. A number of 1,000 multivariate normal samples of $n = 105$ dissimilarities were sampled using the binomial distribution to ensure positive dissimilarity values that follow a normal distribution. The details of the method of sampling from the binomial distribution are described in Frank & Heiser (in press *a*). The Kuhn-Tucker test statistic (Equation 3.16) was calculated for each data set, and the proportion of the replicates in which the value of the test statistic exceeded the value obtained for the original data represents the significance level of the test.

Simulation for nominal standard errors with a priori tree topology

The purpose of this simulation study is to evaluate the performance of the nominal standard errors of the ICLS estimator compared to empirical (bootstrap) standard errors, for the situation where the tree topology is known in advance. In addition, the performance of these nominal standard errors are evaluated by comparing the coverage of the nominal confidence intervals with the coverage of bootstrap confidence intervals . The coverage is equal to the proportion of times the true value is included in the confidence interval.

In this simulation study the performance of the standard errors of the ICLS estimator was evaluated using positive true feature parameters, which represents a situation where it is correct to apply constraints and consequently, the asymptotic properties of the ICLS estimator are expected to hold. For the asymptotic properties to hold, normally distributed errors and homogeneous variances are required as well. Given positive true feature parameters, true distances can be computed that can be used as population values from which dissimilarities can be sampled by adding some error to the true distances. True distances were computed with:

$$\mathbf{d} = \mathbf{X}\boldsymbol{\eta}, \tag{3.25}$$

where the true parameters are equal to the ICLS estimates ($\hat{\eta}_{\text{ICLS}}$) in Table 3.2 and $\mathbf{X}$ is obtained with the feature matrix of the *kinship* data (Figure 3.5). The true tree is starlike because several branches have branch lengths equal to zero. A number of $S = 1,000$ samples of $n = 105$ dissimilarities each, was created by sampling from the binomial distribution and with a homogeneous variance condition created with error variance σ^2 equal to 14.4, which corresponds to the observed residual error variance after fitting the FNM on the original *kinship* data (see for details on the method of binomial sampling, Frank & Heiser, in press *a*). Each simulation sample formed the starting point for a bootstrap of $B = 10,000$ bootstrap samples, using the method of multivariate sampling, which means that for each dissimilarity δ_l ($l = 1, \cdots, n$) sampled from the *kinship* data, the corresponding row of the original $\mathbf{X}$ matrix with features was sampled as well. The simulation yielded 1,000 nominal standard errors ($\hat{\sigma}_{\text{ICLS}}$) for the ICLS estimator. The 1,000 bootstraps (each based on 10,000 bootstrap samples) resulted in 1,000 bootstrap standard deviations (sd_B) of the ICLS estimator.

To evaluate the performance of the estimators, two commonly used measures, the bias and the root mean squared error (*rmse*), were used. Estimates of bias were calculated for the feature parameter estimates $\hat{\eta}_{\text{ICLS}}$, the nominal standard errors $\hat{\sigma}_{\text{ICLS}}$, and the bootstrap standard deviations sd_B. Bias is equal to the expected value of a statistic, $E(\hat{\theta})$, minus the true value θ. For example, the bias of each nominal standard error $\hat{\sigma}_{\text{ICLS}}$ is determined in the simulation study by:

$$bias_{\hat{\sigma}_{\text{ICLS}}} = \left[\frac{1}{S}\sum_{a=1}^{S} \hat{\sigma}_{\text{ICLS}}\right] - \sigma_{\eta}, \tag{3.26}$$

where S indicates the number of simulation samples. The bias of $\hat{\sigma}_{\text{ICLS}}$ is computed with σ_{ICLS} equal to Equation 3.10, using the true values σ^2, $\mathbf{X}$ and $\mathbf{M}$ from Equation 3.12. The bias for the bootstrap standard errors is calculated in the same way, with the exception that $\frac{1}{S}\sum_{a=1}^{S} \hat{\sigma}_{\text{ICLS}}$ is replaced by the sum of the bootstrap standard deviations sd_B.

The *rmse* is equal to the square root of $E[(\hat{\theta} - \theta)^2]$ and takes into account both bias and standard error of an estimate, as can be deduced from the following decomposition (Efron & Tibshirani, 1998):

$$rmse_{\theta} = \sqrt{sd^2_{\hat{\theta}} + bias^2_{\hat{\theta}}}. \tag{3.27}$$

The nominal standard errors ($\hat{\sigma}_{\text{ICLS}}$) were used for the construction of nominal 95% confidence intervals , based on the t distribution ($df = n - T$, with n equal to the number of dissimilarities and T equal to the number of features). Empirical 95% confidence intervals were obtained with the bootstrap-t interval, which is computed in the same way as the nominal confidence interval with the only difference that the bootstrap standard errors (sd_B) are used instead of the estimated standard errors for the sample. For both nominal and empirical confidence intervals , the coverage percentage is equal to the proportion of the simulated samples in which the confidence interval includes the true parameter value.

In a previous study (Frank & Heiser, in press *a*) we also used the *bias-corrected and accelerated* bootstrap interval, the BC_a (Efron & Tibshirani, 1998) in addition to the bootstrap-t interval. Due to the disappointing results obtained for the BC_a intervals, especially when larger numbers of constraints are activated, we restricted this study to the bootstrap-t intervals, which performed better.

Simulation for nominal standard errors with unknown tree topology

The data structure used for this simulation study is based on data from Tversky and Hutchinson (1986, Table 1, p. 5). The data represent mean ratings of similarity between 20 common fruits on a 5-point scale (range 0 - 4, with 4 meaning highly related). For use with an additive tree model, the data were first transformed to dissimilarity values by subtracting each original similarity value from 4. An additive tree was inferred from these dissimilarities with the neighbor-joining (NJ) method (Saitou & Nei, 1987) using the NJ algorithm programmed for Matlab by Strauss (see http://www.biol.ttu.edu/Strauss /Matlab/Matlab.htm). Next, the feature structure (features and internal nodes) was derived from the NJ tree topology by constructing the feature matrix $\mathbf{E}_T$ as in Equation 3.5. The feature matrix equal to the NJ tree topology was submitted to the FNM program (PROXGRAPH) to obtain the ICLS estimates ($\hat{\eta}_{\text{ICLS}}$) for the feature discriminability parameters and the estimated distances. Figure 3.7 shows the resulting tree, where three major clusters become apparent. There is a large cluster with the following three subclusters tropical (exotic) fruit (*coconut, pineapple, pomegranate, banana*), melons (*honeydew, watermelon*) and citrus fruit (*lemon, orange, grapefruit*). This cluster also comprises the tomato that is in a sense exotic because it is not generally recognized as fruit. The second cluster (*grapes, blueberry,*

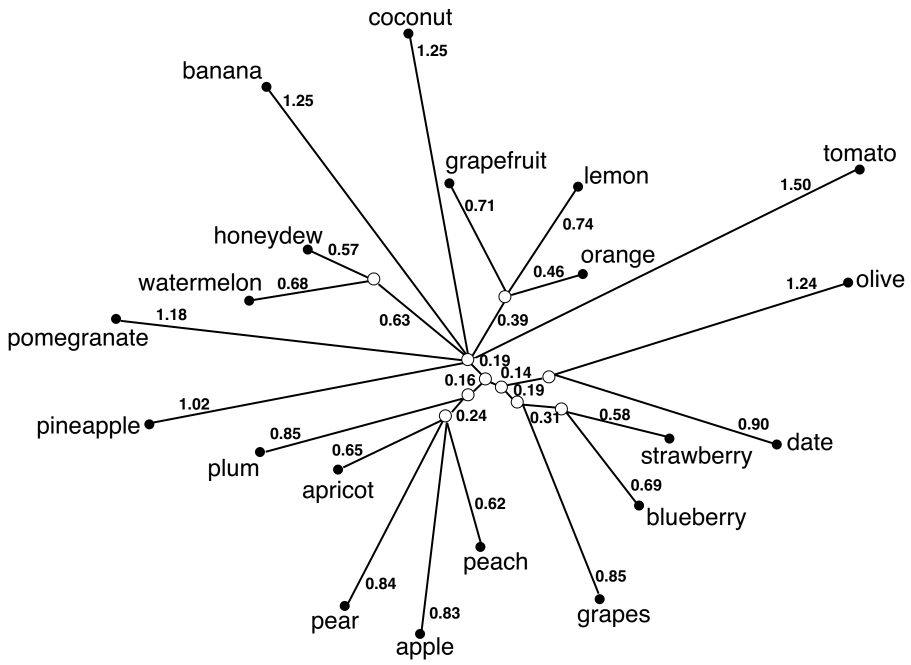

Figure 3.7: Additive tree representation of the *fruit* data obtained with PROXGRAPH based on the tree topology resulting from the neighbor-joining algorithm.

strawberry, date, olive) seems to be determined by the shape and the size of the fruits: small and berry shaped. The third cluster (*plum, apricot, pear, apple, peach*) contains two subfamilies from the rosaceae family, the pome fruits (*pear, apple*) and the stone fruits (*plum, apricot, peach*). The feature structure and the feature discriminability parameters of this tree serve as the true model for the simulation study and are displayed in Table 3.3.

A number of 100 simulation samples with dissimilarities were sampled from the true distances using the aforementioned method of binomial sampling. Two levels of error variance ($\sigma^2 = 0.5, \sigma^2 = 1.0$) were used. To obtain the nominal standard errors when the tree topology is unknown, each sample was divided in a training set and a test set such that the test set contained a proportion of 0.33 of the total sample size. Three levels of total sample size were used: 50, 100 and 300 observations. A total sample size of 50 means that 50 subjects evaluated the relatedness of the 20 fruits on a 5-point scale. The test set contains 33% of the total sample size and the training set the remaining observations. The data that were analyzed were the mean values of the total dissimilarity values in the training set and in the test set. The mean dissimilarity values of the training set of each simulation sample were submitted to the NJ algorithm to obtain an NJ tree topology. Next, the feature structure (features and internal nodes) was derived from this tree topology by constructing the feature matrix $\mathbf{E}_T$ as in Equation 3.5. The feature parameters and associated nominal standard errors were obtained by fitting the training tree topology on the test set dissimilarities using PROXGRAPH. In addition, the prediction error of each sample was estimated with the GCV_{FNM} statistic (Equation 3.24), which was estimated for each test sample using the tree topology obtained in the training sample. The same GCV_{FNM} statistic was also estimated with the training tree topologies and the true distances instead of the test sample dissimilarities. With no sampling error present, the GCV_{FNM} values give an unbiased estimate of the error due to model misspecification. Both GCV_{FNM} estimates were compared in all experimental conditions. The performance of the GCV_{FNM} statistic was further assessed by comparing its distribution in the 6 experimental conditions to the distribution of the number of true features that were recovered in the training tree topologies. Tree topologies that recover a large number of true features should have lower estimates of prediction error.

The performance of the nominal standard errors ($\hat{\sigma}_{\text{ICLS}}$) was evaluated by the coverage proportions of t-confidence intervals constructed with estimates of the nominal standard errors ($\hat{\sigma}_{\text{ICLS}}$). The coverage percentage is equal to the proportion of the simulated samples in which the confidence interval includes the true feature discriminability value, in the same way as for the simulation with fixed tree topology. There is, however, an important difference, because, in this simulation study, an NJ tree topology was estimated for each simulation sample. As a result, the training sample of each simulation sample yielded a feature set that does not necessarily contain all the features present in the true tree topology. Therefore, the proportion of confidence intervals that include the true feature discriminability value can only be obtained for feature discriminability parameters associated with features that are part of the true tree topology. In practice, this means that each tree topology inferred for the training samples, was compared to the true tree topology and only the nominal standard errors associated with features that belong to the true model were used to obtain the coverage proportions of the *t*-confidence intervals . The feature discriminability parameters ($\hat{\eta}_{\text{ICLS}}$) belonging to features that are not included in the true tree topology were also evaluated. To verify the assumption that features that are not included in the true model will lead to small $\hat{\eta}_{\text{ICLS}}$ values, *t*-confidence intervals were constructed using the nominal standard errors ($\hat{\sigma}_{\text{ICLS}}$). The proportion of the confidence intervals that contain the value zero provided evidence for the tenability of the aforementioned assumption.

A few words have to be said about the method used to compare the features resulting from the training sample tree topology with the features from the true tree topology. In terms of a feature model, the tree topology consists of a set of features that are binary $(0, 1)$ variables. A binary vector is in fact the binary code representation of an integer. In the same way, features can be considered as unique representations of integers with the number of bits equal to the number of objects (m). Although there are several binary coding systems available, the Gray code system was used because in the context of FNM it proved to be an efficient method to generate the complete set of distinctive features (Frank & Heiser, in press *b*). In this simulation study, the Gray code system was used to derive the unique Gray code rank number for the features in the true tree topology and for the features in the training sample topologies. The Gray code rank numbers were derived using a Matlab transcription by Burkardt (see http://www.csit.fsu.edu/ burkardt/) of the original algorithms for generating Gray codes in Nijenhuis and Wilf (1978). Since the binary feature vectors can be uniquely identified by a Gray code rank number, the comparison between the features of the training tree topologies and the features of the true tree topology amounts to a simple comparison of integers.

Table 3.3: The 17 cluster features (F_1 - F_{17}) and 20 unique features (F_{18} - F_{37}) with associated feature discriminability parameters for the neighbor-joining tree on the *fruit* data.

Feature	Objects	$\hat{\eta}_{\text{ICLS}}$
F_1	watermelon, honeydew	0.634
F_2	strawberry, blueberry	0.309
F_3	orange, lemon	0.116
F_4	orange, grapefruit, lemon	0.386
F_5	date, olive	0.300
F_6	grapes, strawberry, blueberry	0.192
F_7	pineapple, coconut	0.159
F_8	apple, pear	0.102
F_9	peach, apricot	0.245
F_{10}	peach, apricot, plum	0.038
F_{11}	grapes, strawberry, blueberry, date, olive	0.136
F_{12}	orange, grapefruit, lemon, watermelon, honeydew	0.004
F_{13}	F_{12} + pineapple, coconut	0.095
F_{14}	F_{13} + pomegranate	0.063
F_{15}	apple, peach, pear, apricot, plum	0.155
F_{16}	F_{14} + tomato	0.019
F_{17}	F_{16} + F_{11}	0.009
F_{18}	orange	0.461
F_{19}	apple	0.832
F_{20}	banana	1.253
F_{21}	peach	0.615
F_{22}	pear	0.838
F_{23}	apricot	0.645
F_{24}	plum	0.850
F_{25}	grapes	0.846
F_{26}	strawberry	0.576
F_{27}	grapefruit	0.709
F_{28}	pineapple	1.023
F_{29}	blueberry	0.694
F_{30}	watermelon	0.682
F_{31}	honeydew	0.568
F_{32}	pomegranate	1.179
F_{33}	date	0.895
F_{34}	coconut	1.247
F_{35}	tomato	1.506
F_{36}	olive	1.235
F_{37}	lemon	0.739

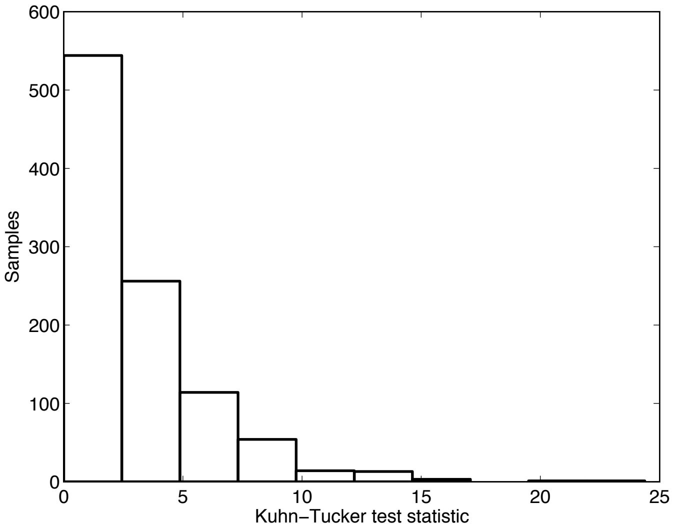

Figure 3.8: Histogram of Kuhn-Tucker test statistic obtained with parametric bootstrap (1,000 samples) with ICLS as H_0 model, based on *kinship* data. The empirical p-value is equal to .74 and represents the proportion of samples with values on the Kuhn-Tucker statistic larger than 0.89, the value of the statistic observed for the sample.

3.6 Results simulation

Results Kuhn-Tucker test and estimates of prediction error

Figure 3.8 shows the result of the simulation based on the additive tree model obtained on the *kinship* data. The Kuhn-Tucker test statistic for the original sample is equal to 0.89 and a proportion of 0.74 of the 1,000 simulated samples have values equal or larger to the sample value of the statistic under the H_0. Therefore, there is no reason to reject the null hypothesis and consequently, it seems appropriate to apply the positivity constraints on these data.

Concerning the estimates of prediction error, the resolved tree yields a GCV_{FNM} value equal to 278.37 and the unresolved tree has $GCV_{\text{FNM}} = 246.10$. Only relative magnitudes of this statistic are meaningful and the conclusion is that the unresolved tree has less prediction error. In summary, the result of the Kuhn-Tucker test shows that the inequality constraints reasonably fit the data, and the estimate of prediction error shows that the unresolved tree has better prediction properties.

Performance of the nominal standard errors for known tree topology

Figure 3.9 shows the mean, the bias, and the *rmse* of the distribution of the 1,000 nominal standard errors as well as the distribution of the 1,000 bootstrap standard errors plotted against the true variability values. Plotting against the true variability

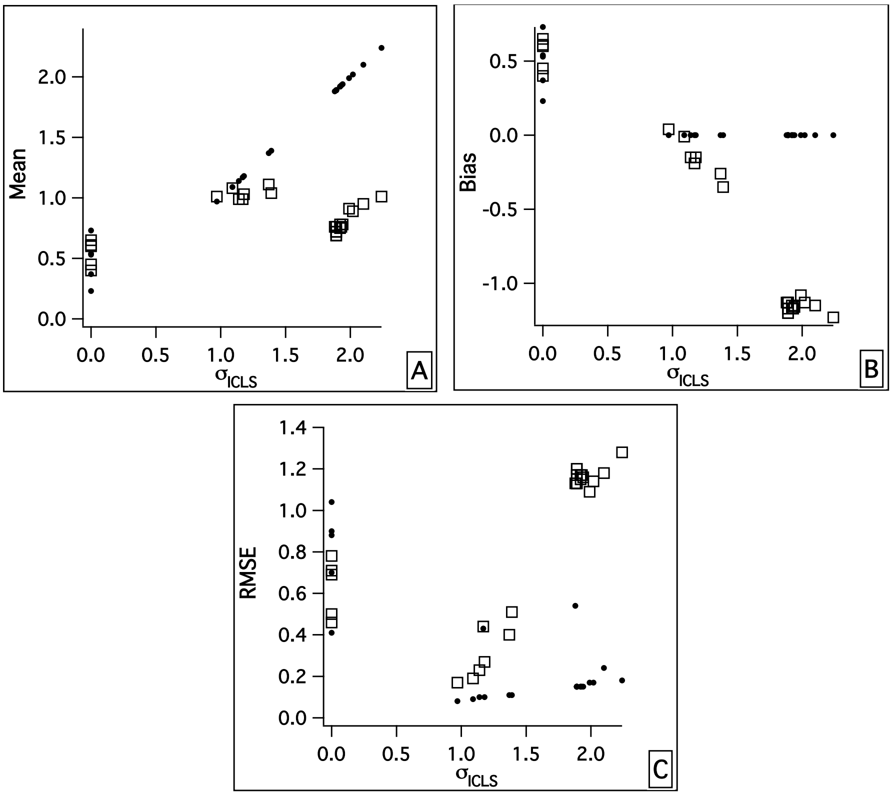

Figure 3.9: Mean (panel A), bias (panel B), and *rmse* (panel C) of the 1,000 simulated nominal standard errors $\hat{\sigma}_{\text{ICLS}}$ (•) and the 1,000 bootstrap standard deviations sd_B (□) plotted against the true nominal standard errors σ_{ICLS}.

allows for comparing the results for the parameters with activated constraints (nominal standard errors equal to zero) and the remaining parameters with no activated constraints. The distribution of the bootstrap standard deviations and the nominal standard errors show a different pattern depending whether constraints are activated or not. When constraints are activated, the pattern of the nominal standard errors is almost equal to the pattern of the bootstrap standard deviations: the values of the mean (panel A), the bias (panel B) and the *rmse* (panel C) are very related. When constraints are not activated, the distribution of the bootstrap standard deviations reveals a clearly different pattern compared to the nominal standard errors. The mean of the bootstrap standard deviations (panel A) is evidently smaller than the mean of the nominal standard errors, which are very close to the true variability

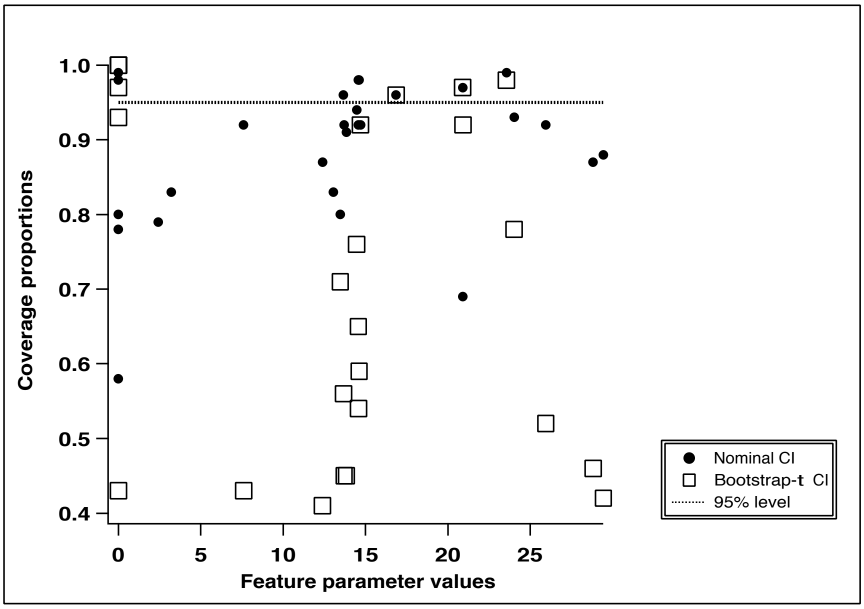

Figure 3.10: Coverage proportions of the nominal *t*-CI and bootstrap *t*-CI for the true feature discriminability values, based on the 1,000 simulated samples.

values, and, consequently, the bootstrap standard deviations are biased downwards (panel B), showing an underestimation of the true variability, whereas the nominal standard errors show almost no bias. The larger bias values for the bootstrap standard deviations, combined with larger variability (not shown) lead to larger values for the *rmse* (panel C).

Figure 3.10 shows the coverage proportions of the nominal and the bootstrap 95% *t*-confidence intervals for the ICLS estimator. The coverage of the nominal confidence intervals is closer to the nominal 95% level than the coverage of the bootstrap confidence intervals that are mostly lower than the nominal values and achieve several low coverage values around 40%. This finding corresponds with the patterns observed in Figure 3.9, where the bootstrap standard deviations are clearly biased downwards.

Performance of the nominal standard errors for unknown tree topology

The right panel of Figure 3.11 displays the distribution of the number of true cluster features present in the NJ tree topologies inferred for the 100 training samples in each experimental condition. Since the unique features are always the same for each topology, only the cluster features are represented. There are 17 cluster features in the true tree topology for the simulation study. The NJ tree topologies obtained in the training samples consistently had 17 cluster features, with two exceptions only. In

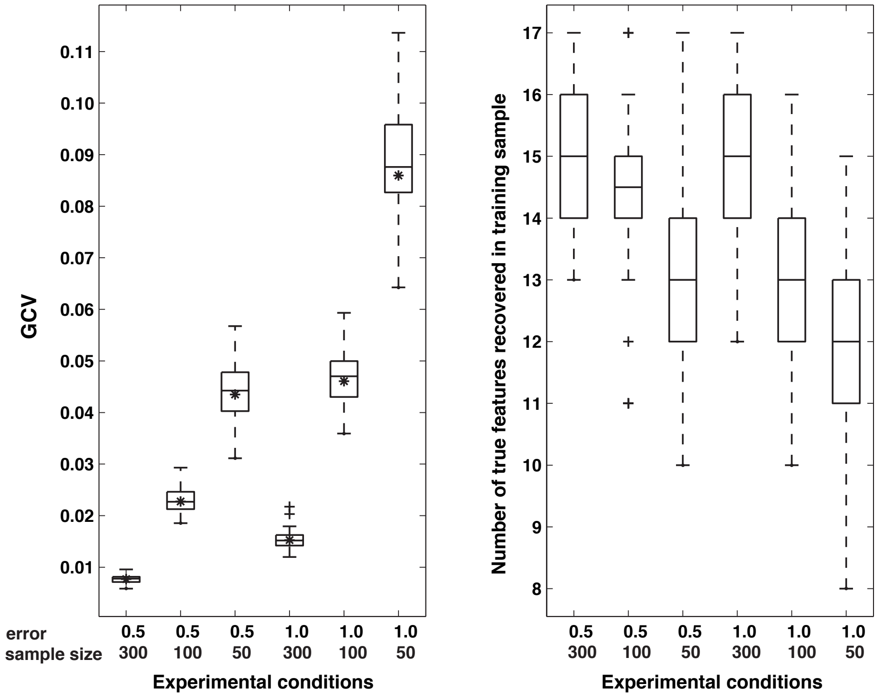

Figure 3.11: *Left panel*: Distribution of the GCV_{FNM} statistic estimated on the test samples based on the tree topology inferred for the training samples under all experimental conditions for 100 simulation samples. The asterisk in each box represents the mean of the true GCV_{FNM} values. *Right panel*: Distribution of the number of cluster features equal to the true cluster features ($T_C = 17$) present in the tree topologies obtained for the training samples of the same 100 simulation samples in each experimental condition.

the low error condition with sample size 100 and with sample size 50, 1 sample out of 100 yielded a NJ tree topology with 16 cluster features. The boxplots in the right panel of Figure 3.11 show that the number of true features recovered in the training sample decreases when the sample size decreases and when the error level is higher, except for sample size 300. The distributions of the prediction error, estimated with the GCV_{FNM} statistic on each test sample (left panel of Figure 3.11), mirror these effects: higher levels of GCV_{FNM} correspond to less well recovered tree topologies. To evaluate the performance of the GCV_{FNM}, this statistic was also estimated with the training tree topology fitted on the true distances. The mean of these GCV_{FNM} values in each of the experimental conditions is represented with an asterisk in the left panel of Figure 3.11 and it is clear that the mean of the GCV_{FNM} values in the test samples is very close to the mean of the GCV_{FNM} values obtained for the true distances.

Table 3.4: Proportion of 95% t-confidence intervals containing the value zero in the test samples for the feature discriminability parameters associated with features not present in the true tree topology

Error level	0.5	0.5	0.5	1.0	1.0	1.0
Sample size	300	100	50	300	100	50
Coverage						
1.00	1.00	0.82	0.85	0.92	0.80	0.81
0.99	0.00	0.18	0.15	0.00	0.15	0.14
0.98	0.00	0.00	0.00	0.08	0.04	0.03
0.96	0.00	0.00	0.00	0.00	0.00	0.02

The feature discriminability values for the features in the training samples that are not included in the true model were recorded for all experimental conditions. Most of these feature discriminability values were equal to zero, but some reached higher values, with a maximum value of 0.26. However, most of these values did not significantly differ from zero, as can be deduced from the coverage proportion of the t-intervals in Table 3.4. In general, at least 96% of the confidence intervals contained the value zero. When error level was low and sample size was equal to 300, all confidence intervals contained the value zero. With increasing error level and decreasing sample sizes, the proportion of confidence intervals spanning zero gradually drops off to 0.96. These results lead to the conclusion that, in general, the feature discriminability parameters associated with features that are not part of the true tree topology, had values that do not significantly differ from zero. Even in the worst case, only a very small proportion (4%) of the feature discriminability parameters associated with features that are not part of the the true tree topology, had values that differ significantly from zero.

Figure 3.12 gives insight in the performance of the nominal standard errors in each experimental condition related to the proportion of correctly recovered features in the training samples. The squares indicate the proportion of features in the NJ tree topologies inferred for the training samples that correspond to the features in the true tree topology. The set of unique features (corresponding to the numbers 18 to 37 in Figure 3.12) is by definition part of the tree topology and therefore, these features have perfect recovery results in all experimental conditions. The recovery of the cluster features is clearly affected by the experimental conditions. The set of features that are less well recovered form the following subset $\{F_3, F_7, F_8, F_{10}, F_{12}, F_{13}, F_{14}, F_{16}, F_{17}\}$. When sample size decreases and error becomes higher, an increasing number of features from this set are less well recovered. It is, however, not surprising that this particular set of features is not well recovered because these features have the smallest feature discriminability parameters in the total feature set (see, Table 3.3). From the point of view of interpretation, these less well recovered cluster features form subsets of fruits that are counterintuitive, like, for example, the combination of citrus fruits and the two types of melons, represented by F_{12} (Table 3.3).

The bullets in Figure 3.12 represent the proportion of nominal t-confidence intervals in the test samples that cover the true feature discriminability parameter for the features that are part of the true tree topology. The feature discriminability parameters that have lower coverage proportions are associated with the same subset of features that are less well recovered. The coverage proportions of the nominal t-confidence intervals are adequate (ranging from 0.95 to 1.0) for the features that are well recovered, but become lower (sometimes reaching values lower than .40) for the features that are less well recovered.

3.7 Discussion

This paper showed how to obtain theoretical standard errors and confidence intervals for the estimates of branch lengths in psychometric additive trees for a priori known tree topologies as well as for estimated tree topologies. The statistical inference theory proposed here derives from the multiple regression framework, which is directly related to the feature representation of additive trees. Using features along with the univariate multiple regression framework offers a different perspective on statistical inference in psychometric additive trees and might be useful for the phylogenetic tree domain as well.

However, a comparison between evolutionary trees and psychometric trees is not straightforward because different assumptions are made about the estimated distances in the tree, and, consequently, the results might not be exchangeable between the two types of tree models. In phylogenetic trees, the distances in the tree represent evolutionary distances, which in most cases are equal to the number of nucleotide substitutions for all pairs of nucleotide sequences representing the species. In psychometrics there is no generally accepted theory about the underlying distribution of dissimilarities between objects. In multidimensional scaling theory, several possible distributions have been proposed. Ramsay (1982) suggested the normal distribution, the log-normal distribution (because of the nonnegative nature of dissimilarities) and a symmetric alternative, the inverse Gaussian (or Wald) distribution. Restle (1961) proposed the gamma distribution and Takane (1981) and Takane and Carroll (1981) used various distributions that take into account the specific data generation process that underlies each data collection method.

Despite these differences, both types of tree domains share the following important property: from evolutionary perspective, but also in psychology, a tree with negative branch lengths has no meaning and cannot be accurate by definition. Consequently, all tree searching algorithms search for tree topologies with positive branch lengths while discarding all tree topologies that yield negative estimates of branch lengths. Searching for a tree with positive branch lengths implies that positivity constraints should be imposed on the estimates of the branch lengths. Imposing inequality constraints during estimation has consequences for the statistical properties of the estimates: they become biased because their distribution is truncated at zero. The presence of the inequality constraints cannot be ignored and should be part of the tree searching algorithms, as already pointed out by Gascuel and Levy (1996), but also when the variability of the branch lengths are estimated.

This paper shows that the theoretical standard errors for inequality constrained least squares estimates are useful in assessing the variability of the branch lengths in psychometric additive trees. For a priori known tree topologies the theoretical standard errors perform well. When the tree topology is not known in advance and estimated with the NJ method, the performance of the confidence intervals based on the theoretical standard errors is adequate, except for the features that have very small feature discriminability values and, at the same time, are not well recovered by the NJ method.

The results of this study are however limited to the normal distribution assumption, necessary in the inequality constrained least squares framework. In addition, the assumption of homogeneous variances (Equation 3.7) is arguable because the dissimilarity values that share the same objects are likely to be correlated. In multidimensional scaling, solutions have been proposed by Ramsay (1982) who uses a multiplicative variance components model instead of the additive model of Equation 3.7 and introduces MINQUE variance estimates for cases where the configuration matrix is known. In the phylogenetic domain, Bulmer (1991) uses generalized least squares to account for the heterogeneity. In theory, the combination of ICLS estimates with generalized least squares, yielding inequality constrained generalized least squares estimates (ICGLS), could be a solution. In practice, the statistical properties are barely known (Werner, 1990; Werner & Yapar, 1996) and since the generalized least squares estimates become a computational burden with large number of objects (*cf.* Felsenstein, 2004), the ICGLS estimates are a difficult way to go.

The use of features as predictor variables in a multiple regression framework has additional advantages. An important advantage is that measures of prediction error that are regularly used in this framework become easily available. In this paper we showed how the statistical inference theory of the inequality constrained least squares estimator can be incorporated in the general theory of linear fitting methods to obtain an estimate of prediction error, the *generalized cross-validation* statistic, which is a convenient closed form formula that approximates leave-one-out cross-validation. Besides the very low computational costs, another advantage of the *generalized cross-validation* statistic is that it can be used to compare different tree topologies with the same number of degrees of freedom, i.e. models that have the same number of predictors or features. For the likelihood ratio test, a commonly used test to compare phylogenies, the comparison of tree topologies with the same degrees of freedom is a problem because the test is limited to the case of nested topologies (*cf.* Felsenstein, 2004, Chapter 19).

Another advantage of considering the feature framework for additive trees, is that a frequently used test in the phylogeny domain, testing speciation or population splitting, can be done explicitly by adding *cluster features* to the model. In phylogenetic trees, internal nodes are usually called branching points and indicate that an important event of speciation or population splitting occurred there (*cf.* Nei et al., 1985). The internodal distances are not observed and therefore are inferred form the other, non-internodal distances, as well as the associated standard errors. Several tests for the branching points have been proposed (Bulmer, 1991; Li, 1989; Nei et al., 1985; Tajima, 1992). An alternative for the interior-branch test is the bootstrap method proposed by Felsenstein (1985), which calculates the proportion of bootstrap trees that agree with the original tree topology inferred for the sample. A population splitting that occurs in a large proportion of bootstrap trees is considered to be very plausible. Sitnikova, Rzhetsky, and Nei (1995) compared the interior-branch test with the bootstrap test and concluded that the bootstrap test tends to yield conservative confidence values compared to the interior-branch test and that the difference between the two tests becomes more salient when the true tree is starlike, which means that some branches have length zero resulting from the correction of negative branch lengths.

Considering features in additive trees allows for a different way to test the branching points. In the phylogeny literature, the branching points result from a certain topology that depends on the tree finding algorithm. FNM offers the possibility to test explicitly for specific ancestral species just by adding cluster features to the feature matrix $\mathbf{E}_T$. The values of the feature discriminability parameters and the associated confidence intervals indicate whether the ancestral species are plausible. The simulation study in this paper for the case of unknown tree topologies, inferred for each simulation sample with the NJ method, is in fact a parametric version of Felsenstein's bootstrap method (1985) because it calculates the proportion of features from the true topology that are recovered in the samples while assuming a model with normally distributed error terms. The confidence intervals obtained with the theoretical standard errors for the feature discriminability parameters led to the same conclusion about the most plausible features (including cluster features that indicate speciation) in the model, but at much less computational cost.

Although strong assumptions have to be made (normally distributed errors and homogeneous variances), we believe that the theoretical standard errors for the inequality constrained least squares estimates are useful for estimating the variability of branch lengths of tree topologies obtained with algorithms like ADDTREE and NJ, which use least squares estimates for the branch lengths. Using features along with the multiple regression framework has many advantages, as has been demonstrated in this paper. Nevertheless, the question remains whether these results are useful for the phylogenetic trees. The answer relies on the challenge to combine the theoretical standard errors for the inequality constrained least squares estimator with the many methods proposed in the phylogenetic literature that take into account the way the evolutionary distances were obtained.

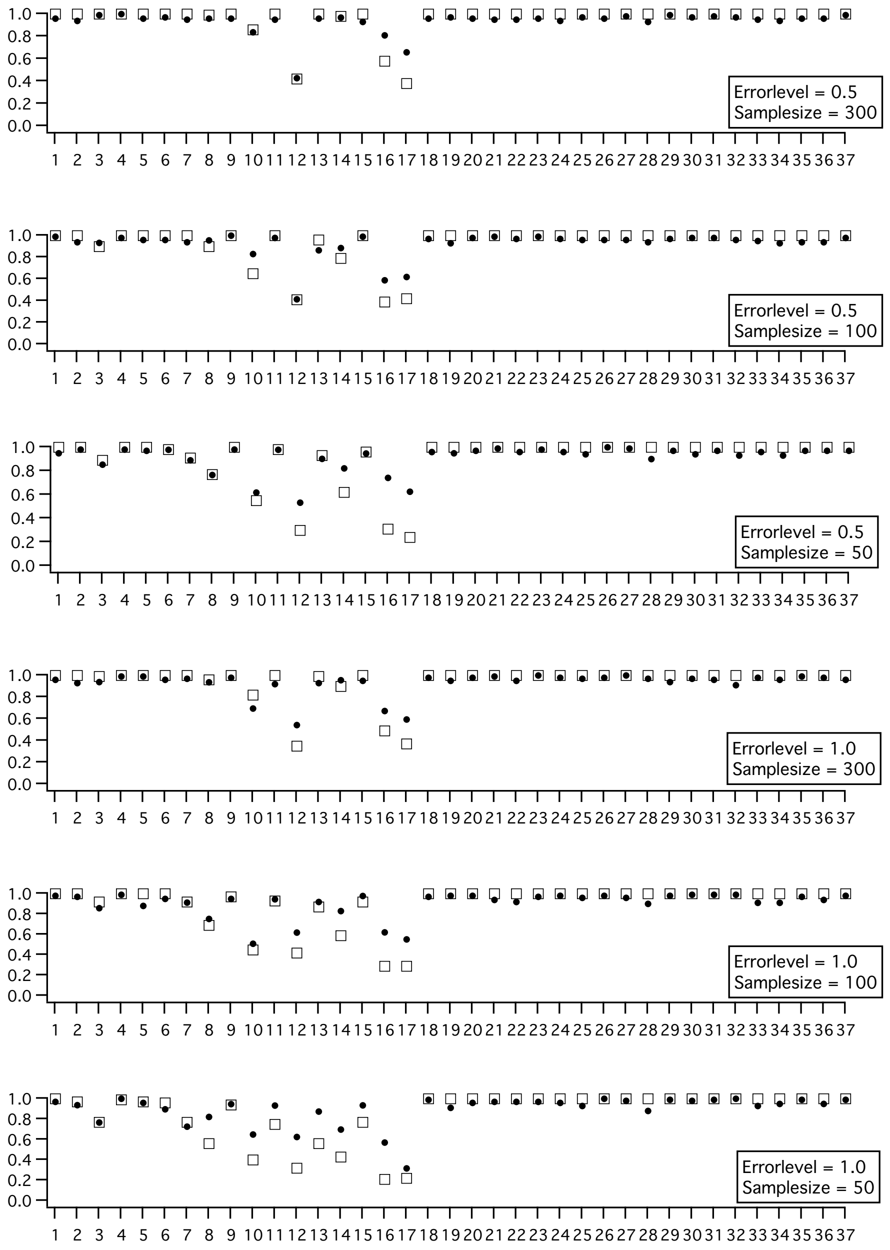

Figure 3.12: Coverage proportions in all experimental conditions for feature discriminability parameters based on nominal *t*-CI (•) in the test samples and proportions recovered true features in the training samples (□) for each of the 37 features forming the true tree topology.