Ensembles of Nested Dichotomies for Hierarchical Classification and Their Application to SCOP

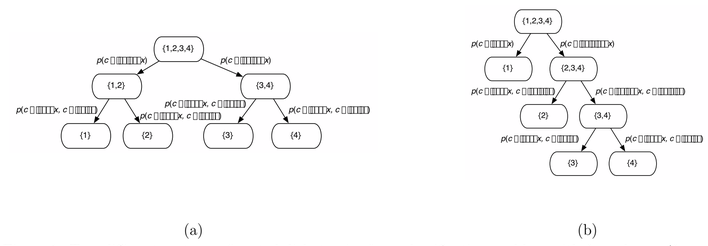
Arthur Zimek
zimek@informatik.uni-muenchen.de
LMU/TU München, Germany

Eibe Frank
eibe@cs.waikato.ac.nz
University of Waikato, New Zealand

Stefan Kramer
kramer@in.tum.de
TU München, Germany

Nested Dichotomies

Nested Dichotomies ([1],[2]) are a standard statistical technique for tackling certain polytomous classification problems with logistic regression. They can be represented as binary trees that, at each node, divide the set of classes C_i associated with the internal node i into two disjoint subsets C_{i_1} and C_{i_2} that taken together contain all the classes in C_i. The nested dichotomies' root node contains all the classes of the corresponding polytomous classification problem. Each leaf node contains a single class (i.e. for an n-class problem, there are n leaf nodes and $n-1$ internal nodes).



(a) (b)

Figure 1: Two different systems of nested dichotomies for a classification problem with four classes (from [2]).

Figure 1 shows two of the 15 possible nested dichotomies for a four-class classification problem. The probability of class 4 for an instance x based on these trees is given by

$$p_a(c=4|x) = p_a(c \in \{3,4\}|x) \times p_a(c \in \{4\}|x, c \in \{3,4\})$$

for tree a, and respectively for tree b:

$$p_b(c=4|x) = p_b(c \in \{2,3,4\}|x) \times p_b(c \in \{3,4\}|x, c \in \{2,3,4\}) \times p_b(c \in \{4\}|x, c \in \{3,4\}).$$

Both trees represent equally valid, albeit different class probability estimators. In general there is no reason to trust one of the estimates more than others. Consequently it makes sense to treat all possible trees as equally likely and form overall class probability estimates by averaging the estimates obtained from different trees.

Ensembles of Nested Dichotomies

The number of possible trees for an n-class problem grows with the double-factorial and is given by the following recurrence relation:

$$T(n) = (2n-3) \times T(n-1), \quad (1)$$
$$T(1) = 1,$$

since there are $(n-1)+(n-2) = 2n-3$ distinct possibilities to add a new class into a tree for $n-1$ classes, one for each of the $n-1$ leaf-nodes and the $n-2$ internal nodes.
For an increasing number of classes it becomes therefore infeasible to consider all possible nested dichotomies. This is true even for problems with a moderate number of classes, as is illustrated in Figure 2.

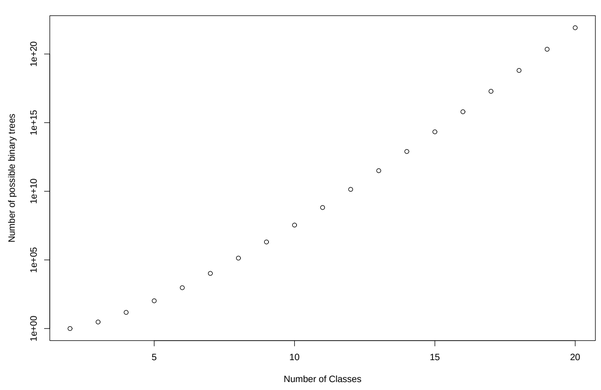


Figure 2: Growth of number of possible binary-trees for a multi-class problem of n classes. Note the logarithmic scale.

Since there is in general no reason to prefer one nested dichotomy over another, a method called 'Ensembles of Nested Dichotomies (ENDs)' was proposed recently ([2]), which takes a random sample from the space of all distinct trees for a given n-class problem and forms class probability estimates for a given instance x by averaging the estimates obtained from the individual ensemble members. The main motivation for taking an ensemble approach is that the variance component of the error can be reduced in this way.

Hierarchical Structure of Classes

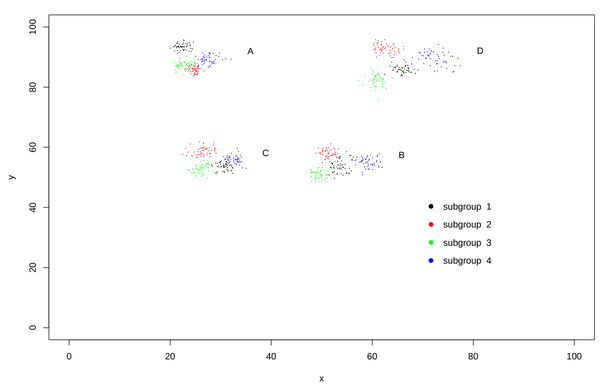
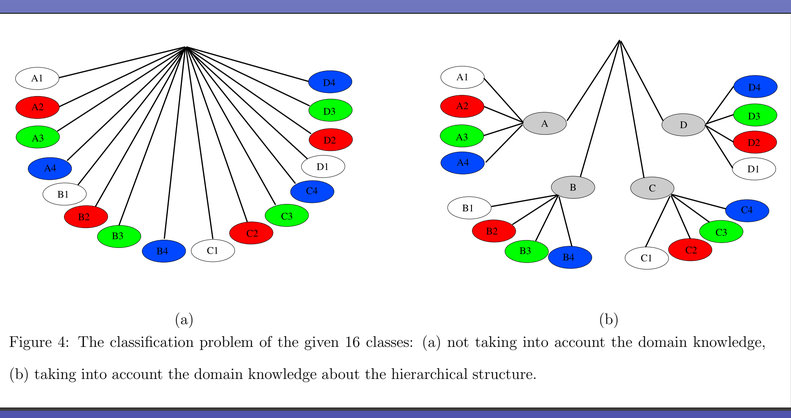


Figure 3: A polytomous classification problem in two-dimensional Euclidean space, consisting of 16 classes, i.e. 4 super-classes each composed of 4 classes.

Not accounting for domain knowledge concerning the 16 classes given in Figure 3 one would describe the classification problem as illustrated in Figure 4a: One classifier ought to learn the classification with respect to all 16 classes.
However, it is obvious that the problem consists of four super-classes, A, B, C, and D, which are easily separable from each other. Each of these four super-classes contains four subclasses (subgroup 1 to 4, respectively), which are not completely separable. But one may expect any classifier to improve its accuracy for a properly represented task, as illustrated in Figure 4b.



(a) (b)

Figure 4: The classification problem of the given 16 classes: (a) not taking into account the domain knowledge, (b) taking into account the domain knowledge about the hierarchical structure.

Also, the size of the space of valid nested dichotomies to choose a random sample from can be considerably reduced by introducing a hierarchical restriction: Whilst there are, e.g., (according to Equation 1) $T(16) = 6.19 \times 10^{15}$ possible nested dichotomies for the problem of Figure 4a, a restriction according to Figure 4b would allow only $T(4)^5 = 759,375$ of them.

Ensembles of Hierarchical Nested Dichotomies

Therefore, we suggest to take into account the hierarchical structure of the data for sampling nested dichotomies. This is straightforward: Considering Figure 3, the 16 classes are obviously organized by an n-ary tree with the original 16 classes as leaves, and four super-classes introduced as internal nodes (see Figure 4b). This n-ary tree can be represented by several binary-trees in a valid manner. Such a valid representation ('binarization') of an n-ary tree will always contain those super-classes as internal nodes that are contained by the n-ary tree. Moreover, it will contain groups of classes that are either super-classes or subclasses of the super-classes occurring in the originally n-ary tree, but no groups that fall in neither of those categories.

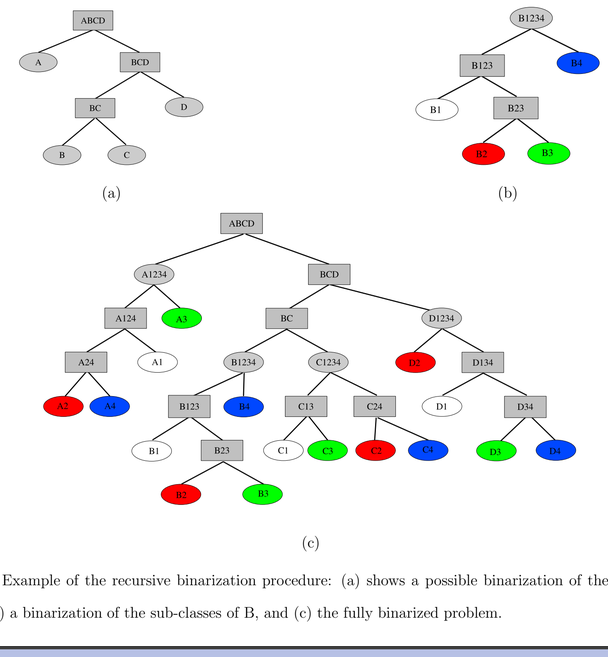


Figure 5: Example of the recursive binarization procedure: (a) shows a possible binarization of the four super-classes, (b) a binarization of the sub-classes of B, and (c) the fully binarized problem.

For the data of Figure 3, which has a pronounced hierarchical structure, the proposed method of EHNDs improves considerably more on pairwise coupled support vector machines than ENDs (see Table 1).

Method:	SVM	ENDs	EHNDs
Accuracy (Q):	78.25	82.375	94.75

Table 1: Performance of pairwise coupled Support Vector Machines, ENDs and Ensembles of Hierarchical Nested Dichotomies (EHNDs) on the data of Figure 3 with the hierarchy defined by: {{A1,A2,A3,A4}, {B1,B2,B3,B4}, {C1,C2,C3,C4}, {D1,D2,D3,D4}}. The numbers are the percentage of correctly classified instances for a 10-fold stratified cross-validation.

Hierarchical Classification of Proteins

For the hierarchical classification of proteins we used the feature-space and data of Ding and Dubchak ([3], based on [4],[5], adapted to a more recent SCOP-classification (SCOP 1.61, [6]) by Tan et al. [7]) – see Table 2.

Group	Method	Training				Test			
		TPR	FPR	PPV	F1	TPR	FPR	PPV	F1
All-α	PART	0,9359	0,0014	0,9599	0,9478	0,5758	0,0263	0,502	0,5363
	EHND (PART)	1	0	1	1	0,7273	0,0071	0,8076	0,7653
	SVM	0,9872	0,0022	0,9662	0,9766	0,7273	0,0121	0,6655	0,695
	EHND (SVM)	0,9615	0,0028	0,9402	0,9507	0,7576	0,0093	0,8106	0,7832
All-β	PART	0,9274	0,0053	0,9107	0,919	0,3774	0,0352	0,3735	0,3754
	EHND (PART)	1	0	1	1	0,6226	0,0269	0,5945	0,6082
	SVM	0,9355	0,0075	0,8925	0,9135	0,5849	0,0379	0,4896	0,533
	EHND (SVM)	0,9516	0,0052	0,9255	0,9384	0,5849	0,0356	0,5802	0,5825
α/β	PART	0,8873	0,0052	0,9052	0,8962	0,2623	0,0386	undef	undef
	EHND (PART)	1	0	1	1	0,4262	0,0408	0,3604	0,3906
	SVM	0,8451	0,0159	0,8722	0,8584	0,4918	0,0299	0,4942	0,493
	EHND (SVM)	0,9155	0,0121	0,8908	0,903	0,4426	0,0481	0,4062	0,4236
$\alpha+\beta$	PART	0,9688	0,0059	0,9135	0,9403	0,2143	0,0198	0,3571	0,2679
	EHND (PART)	0,9375	0,0006	0,9722	0,9545	0,4286	0,0338	0,3929	0,4099
	SVM	0,625	0,002	0,95	0,754	0,1429	0,006	0,4762	0,2198
	EHND (SVM)	0,625	0	1	0,7692	0,2857	0,0087	undef	undef
Small Proteins	PART	1	0	1	1	0,6923	0	1	0,8182
	EHND (PART)	1	0,0027	0,9697	0,9846	0,9231	0	1	0,96
	SVM	1	0,0053	0,9412	0,9697	0,8462	0,0062	0,9167	0,88
	EHND (SVM)	1	0	1	1	1	0	1	1
Total	PART	0,924	0,0041	0,9254	0,9247	0,3851	0,0308	undef	undef
	EHND (PART)	0,9951	0,0003	0,9954	0,9953	0,5805	0,0266	0,5669	0,5736
	SVM	0,8946	0,0088	0,9078	0,9012	0,5632	0,0253	0,5554	0,5593
	EHND (SVM)	0,9191	0,0063	0,9279	0,9235	0,5747	0,0302	undef	undef

Table 2: Performance of EHNDs (40 ensemble-members) in comparison to their respective base-classifier.
TPR = TP / (TP + FN)
FPR = FP / (FP + TN)
PPV = TP / (TP + FP)
F_1 = (2TPR × PPV)/(TPR + PPV)

References

[1] John Fox. *Applied Regression Analysis, Linear Models, and Related Methods.* Sage, 1997.

[2] Eibe Frank and Stefan Kramer. Ensembles of nested dichotomies for multi-class problems. In *Twenty-first international conference on Machine learning*. ACM Press, 2004.

[3] Chris H. Q. Ding and Inna Dubchak. Multi-class protein fold recognition using support vector machines and neural networks. *Bioinformatics*, 17(4):349–358, 2001.

[4] Inna Dubchak, Ilya Muchnik, Stephen R. Holbrook, and Sung-Hou Kim. Prediction of protein folding class using global description of amino acid sequence. *Proc. Natl. Acad. Sci. USA*, 92:8700–8704, September 1995.

[5] Inna Dubchak, Ilya Muchnik, Christopher Mayor, Igor Dralyuk, and Sung-Hou Kim. Recognition of a protein fold in the context of the SCOP classification. *PROTEINS: Structure, Function, and Genetics*, 35:401–407, 1999.

[6] Loredana Lo Conte, Steven E. Brenner, Tim J. P. Hubbard, Cyrus Chothia, and Alexey G. Murzin. SCOP database in 2002: refinements accomodate structural genomics. *Nucleic Acids Research*, 30(1):264–267, 2002.

[7] Aik Choon Tan, David Gilbert, and Yves Deville. Multi-class protein fold classification using a new ensemble machine learning approach. *Genome Informatics*, 14:206–217, 2003.