Hierarchical Classification
Using
Ensembles of Nested Dichotomies

Arthur Zimek

Reduction of Polytomies to Dichotomies

Motivation:

- application of binary classifiers to multiclass-problems
- simplifying decision-boundaries

Principle:

- create a set of mappings of n classes to 2 classes
- employ a set of binary classifiers each trained for one of the dichotomies

Reduction of Polytomies to Dichotomies

Example:

- four-class problem:

$$C = \{c_1, c_2, c_3, c_4\}$$

- e.g. three mappings $m_i : C \to \{-1, 1\}$:

$$m_1 : c \mapsto \begin{cases} 1 & \text{if} \quad c \in \{c_1, c_2\} \\ -1 & \text{if} \quad c \in \{c_3, c_4\} \end{cases}$$

$$m_2 : c \mapsto \begin{cases} 1 & \text{if} \quad c \in \{c_1, c_3\} \\ -1 & \text{if} \quad c \in \{c_2, c_4\} \end{cases}$$

$$m_3 : c \mapsto \begin{cases} 1 & \text{if} \quad c \in \{c_2, c_3\} \\ -1 & \text{if} \quad c \in \{c_1, c_4\} \end{cases}$$

- Resulting decomposition matrix:

$$\begin{pmatrix} 1 & 1 & -1 \\ 1 & -1 & 1 \\ -1 & 1 & 1 \\ -1 & -1 & -1 \end{pmatrix}$$

(one row per class, one column per mapping resp. per classifier)

Reduction of Polytomies to Dichotomies

Possibilities:

- *one-versus-rest*:

$$\begin{pmatrix} 1 & -1 & -1 & -1 \\ -1 & 1 & -1 & -1 \\ -1 & -1 & 1 & -1 \\ -1 & -1 & -1 & 1 \end{pmatrix}$$

- *all-pairs*:

$$\begin{pmatrix} 1 & 1 & 1 & 0 & 0 & 0 \\ -1 & 0 & 0 & 1 & 1 & 0 \\ 0 & -1 & 0 & -1 & 0 & 1 \\ 0 & 0 & -1 & 0 & -1 & -1 \end{pmatrix}$$

- *minimal*:

$$\begin{pmatrix} 1 & 1 \\ 1 & -1 \\ -1 & 1 \\ -1 & -1 \end{pmatrix}$$

- *complete*:

$$\begin{pmatrix} 1 & -1 & 1 & -1 & 1 & -1 & 1 \\ -1 & 1 & 1 & -1 & -1 & 1 & 1 \\ -1 & -1 & -1 & 1 & 1 & 1 & 1 \\ -1 & -1 & -1 & -1 & -1 & -1 & -1 \end{pmatrix}$$

- *random (ECOC)*:

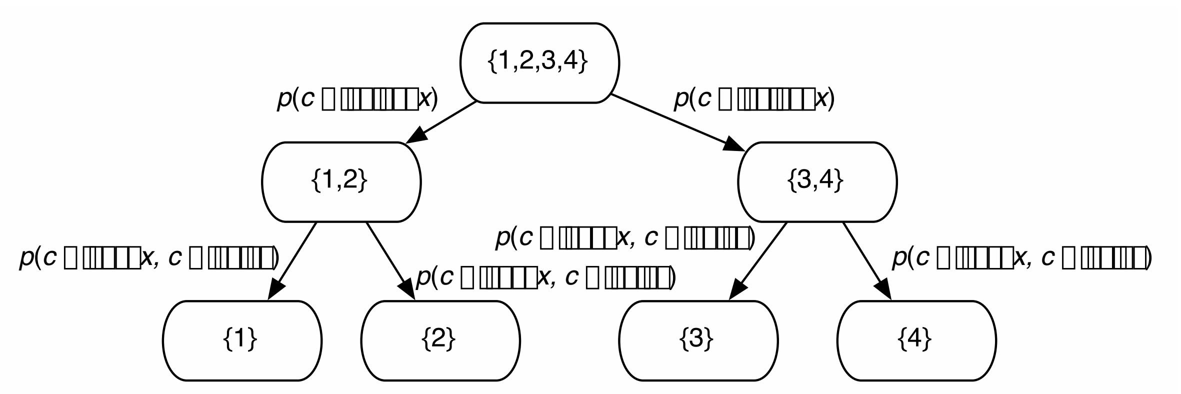
$$\begin{pmatrix} 1 & 1 & -1 \\ 1 & -1 & 1 \\ -1 & 1 & 1 \\ -1 & -1 & -1 \end{pmatrix}$$

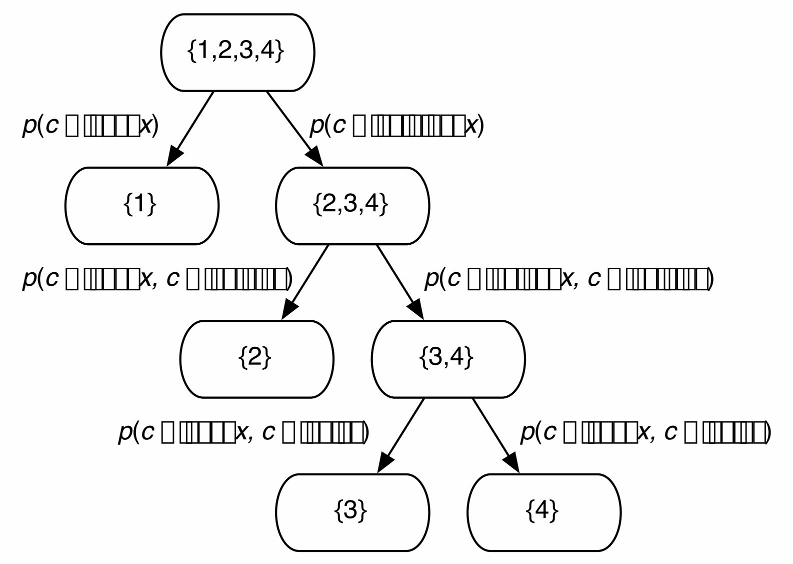
- *nested dichotomies*:

$$\begin{pmatrix} 1 & 1 & 0 \\ 1 & -1 & 0 \\ -1 & 0 & 1 \\ -1 & 0 & -1 \end{pmatrix}$$

Nested Dichotomies

$$\begin{pmatrix} 1 & 1 & 0 \\ 1 & -1 & 0 \\ -1 & 0 & 1 \\ -1 & 0 & -1 \end{pmatrix} \qquad \begin{pmatrix} 1 & 0 & 0 \\ -1 & 1 & 0 \\ -1 & -1 & 1 \\ -1 & -1 & -1 \end{pmatrix}$$





Properties of Nested Dichotomies

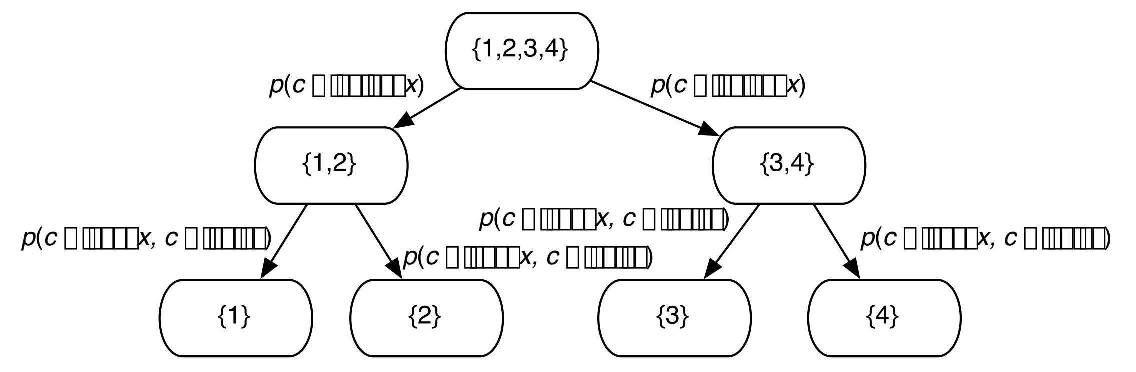
- The dichotomies are independent, thus class probability estimation is derived by multiplication along a branch.

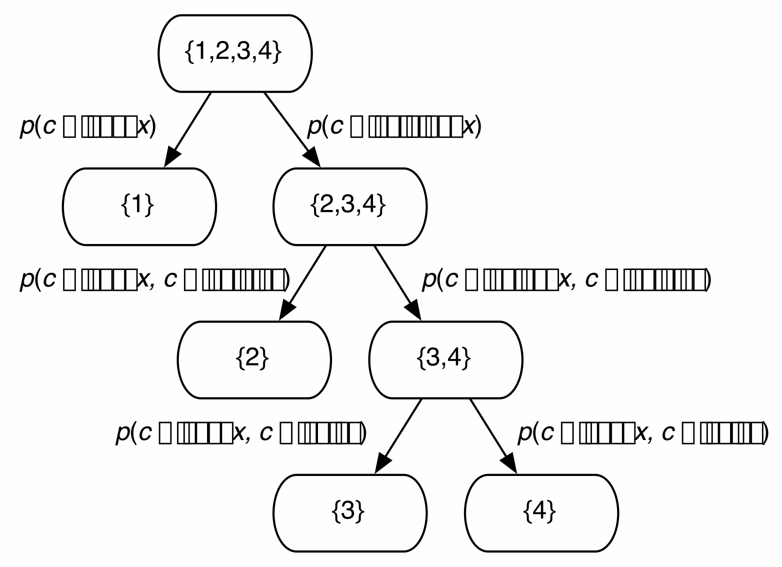
$$\begin{aligned} p(c=m|x) \;=\; & \prod_{i=1}^{n-1} \big(I(m \in C_{i_1})\, p(c \in C_{i_1}|x,\, c \in C_i) \;+ \\ & \quad I(m \in C_{i_2})\, p(c \in C_{i_2}|x,\, c \in C_i)\big) \end{aligned}$$

- Any system of nested dichotomies is biased by imposing a certain order on the set of classes.

Properties of Nested Dichotomies

The class probability estimations will usually differ for two different systems of nested dichotomies:





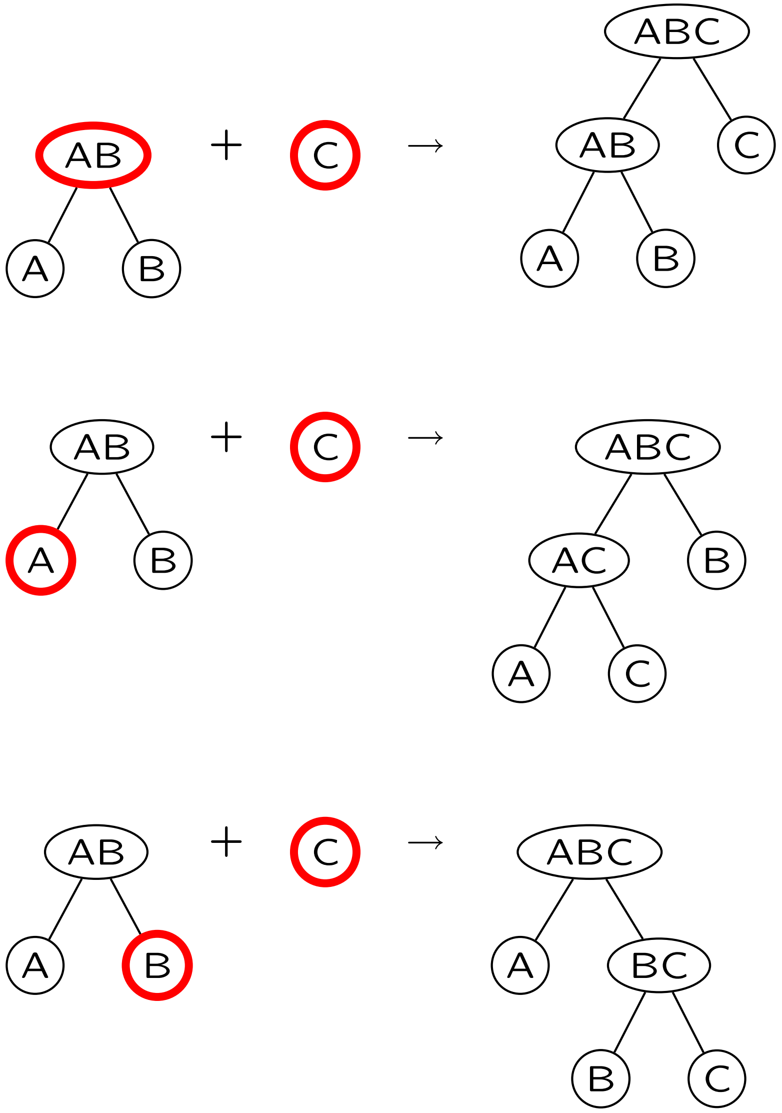
$$\begin{aligned} p(c = 4|x) &= p(c \in \{3,4\}|x) \times \\ &\quad\; p(c \in \{4\}|x, c \in \{3,4\}) \end{aligned}$$

$$\begin{aligned} p(c = 4|x) &= p(c \in \{2,3,4\}|x) \times \\ &\quad\; p(c \in \{3,4\}|x, c \in \{2,3,4\}) \times \\ &\quad\; p(c \in \{4\}|x, c \in \{3,4\}) \end{aligned}$$

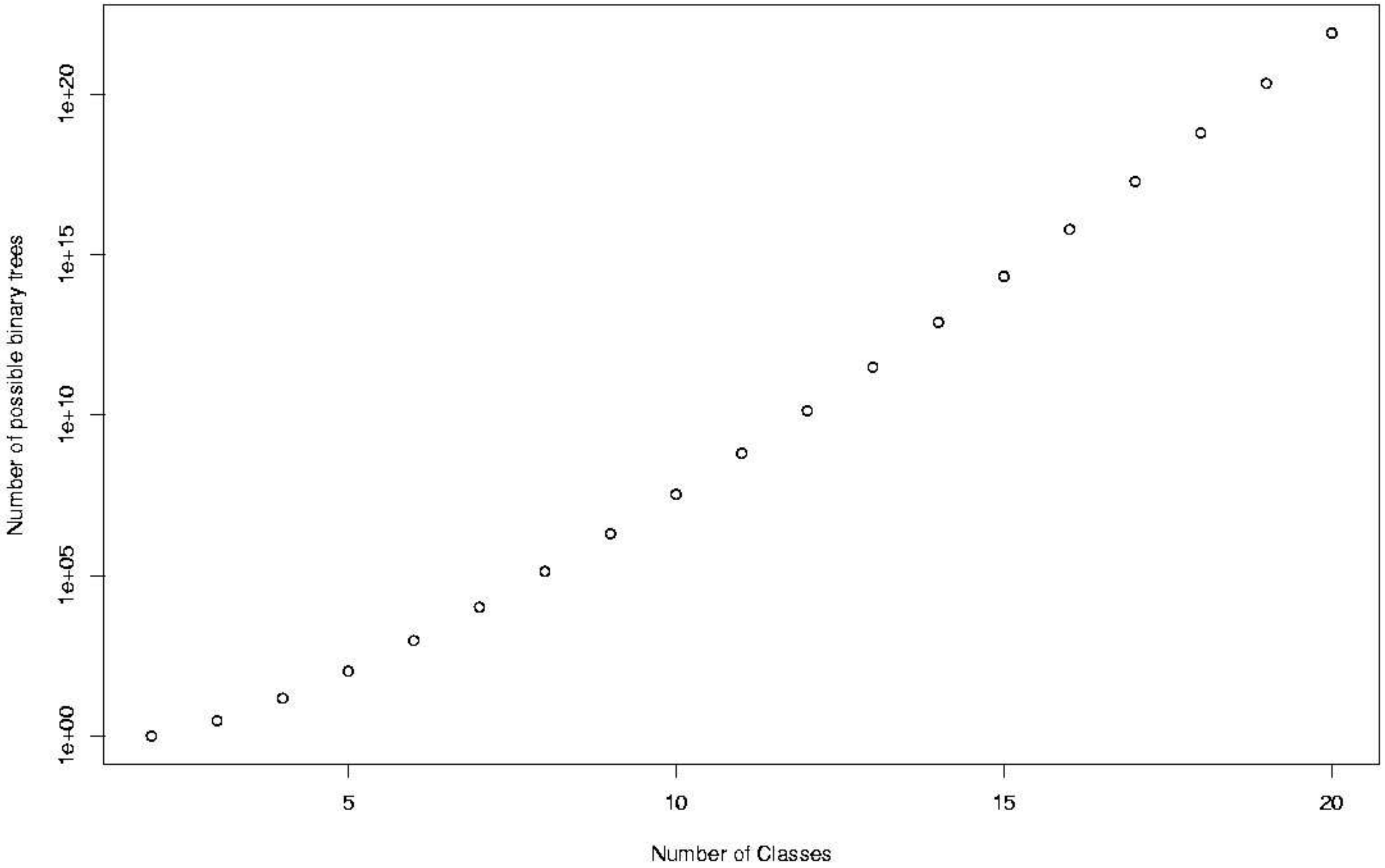
How to build Nested Dichotomies

```
PROCEDURE insert(Class, Index, Tree)
{
    Subtree ← subtree of Tree at Index;
    Replace node Index in Tree with (Subtree, Class);
    RETURN Tree;
}

PROCEDURE $f_{ND}$(ClassList)
{
    IF length(ClassList) $< 3$ THEN
        RETURN ClassList;
    ELSE
        (First, Second|RestList) ← ClassList;
        Tree ← (First, Second);
        FOR $i$ ← 3 TO length(ClassList) DO
            NextClass ← $i^{th}$ element of ClassList;
            Index ← random number $r : 0 < r < 2i - 3$;
            Tree ← insert(NextClass, Index, Tree);
        RETURN Tree;
}
```



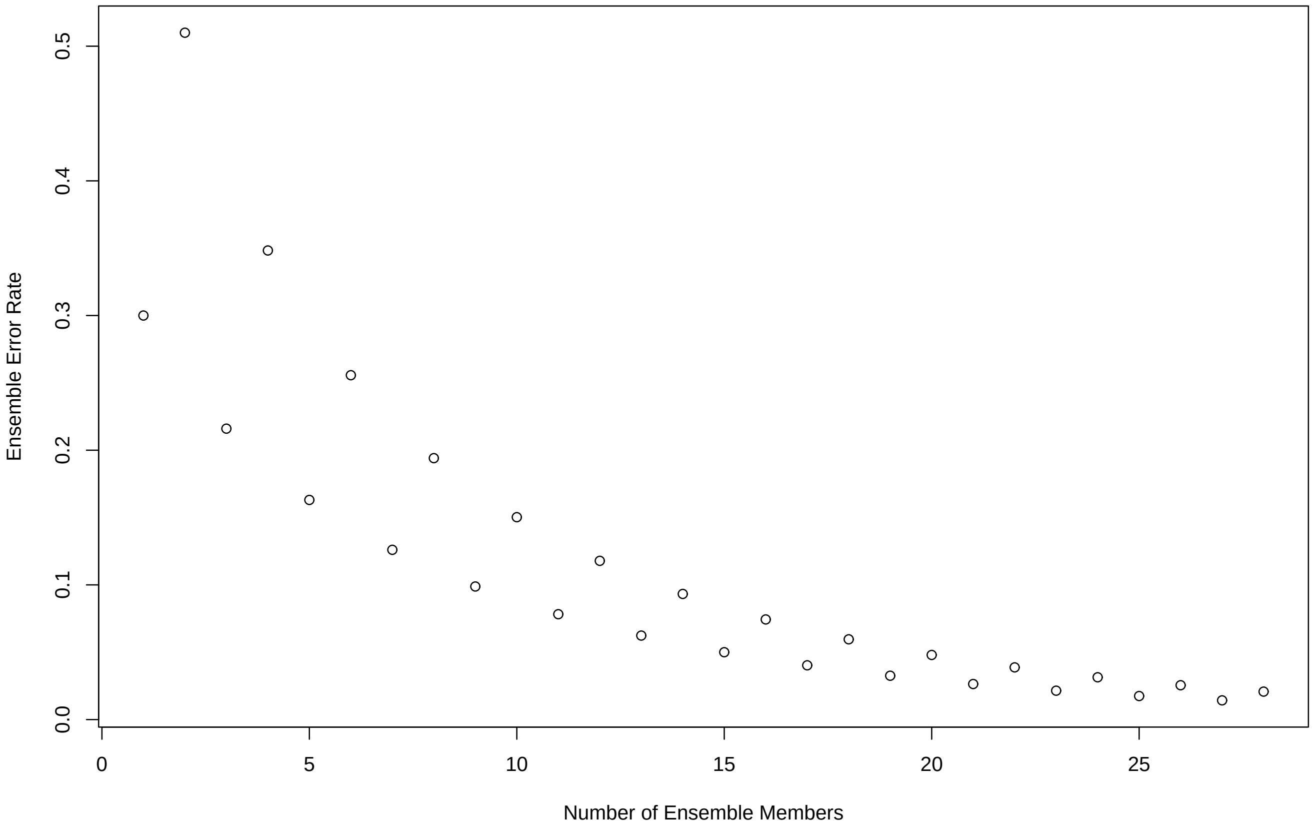
Number of possible NDs



$$
\begin{aligned}
T(n) &= (2n-3) \times T(n-1) \\
T(1) &= 1
\end{aligned}
$$

Why to build Ensembles

The ensemble will only be wrong if at least 50% of its members are wrong:



$$\bar{p}\left(k,p\right) \;=\; \sum_{i=\lceil k/2 \rceil}^{k} \binom{k}{i} p^i (1-p)^{k-i}$$

Ensembles of Nested Dichotomies

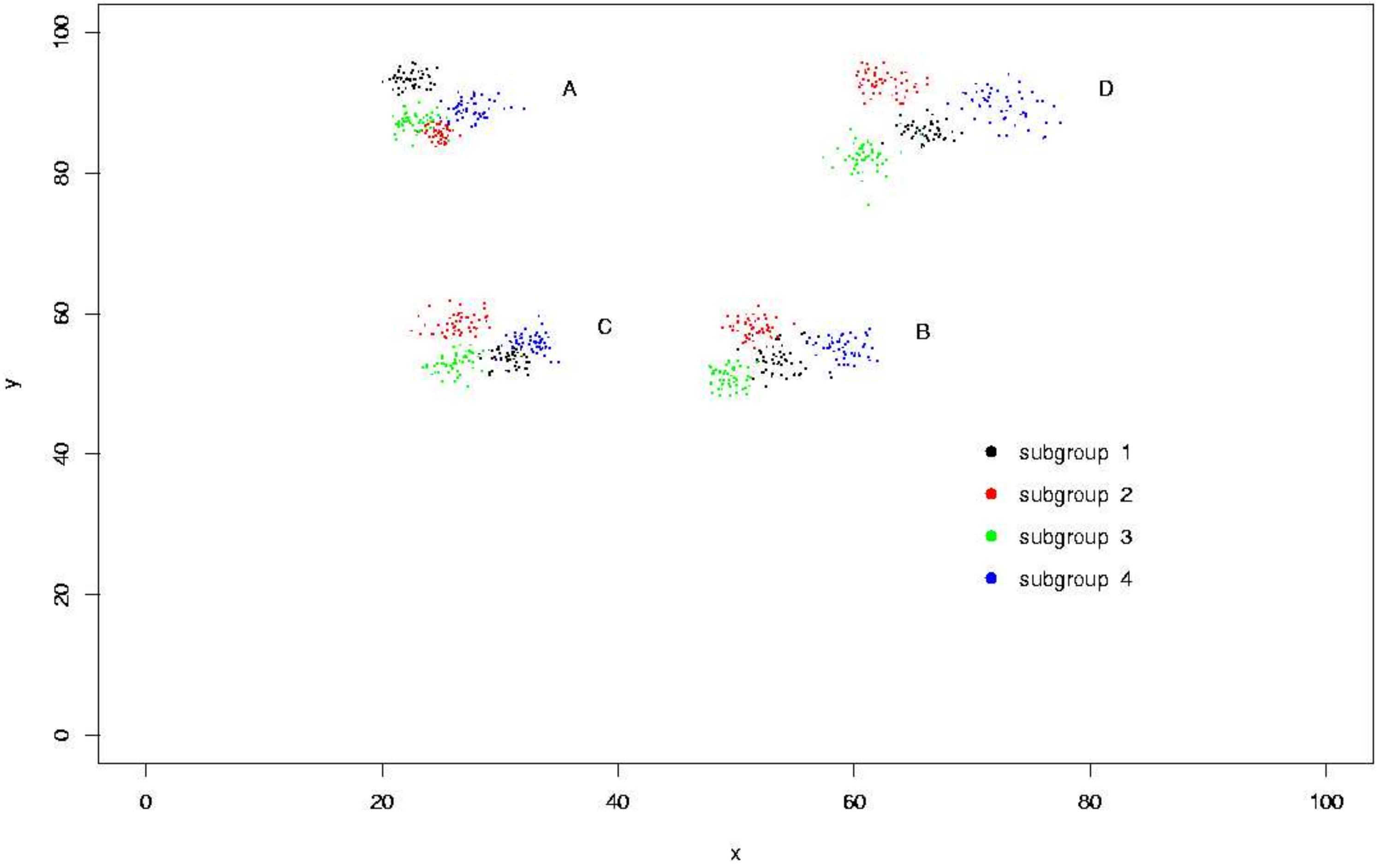
- reduction of error variance by building an ensemble

- random sampling from space of possible NDs

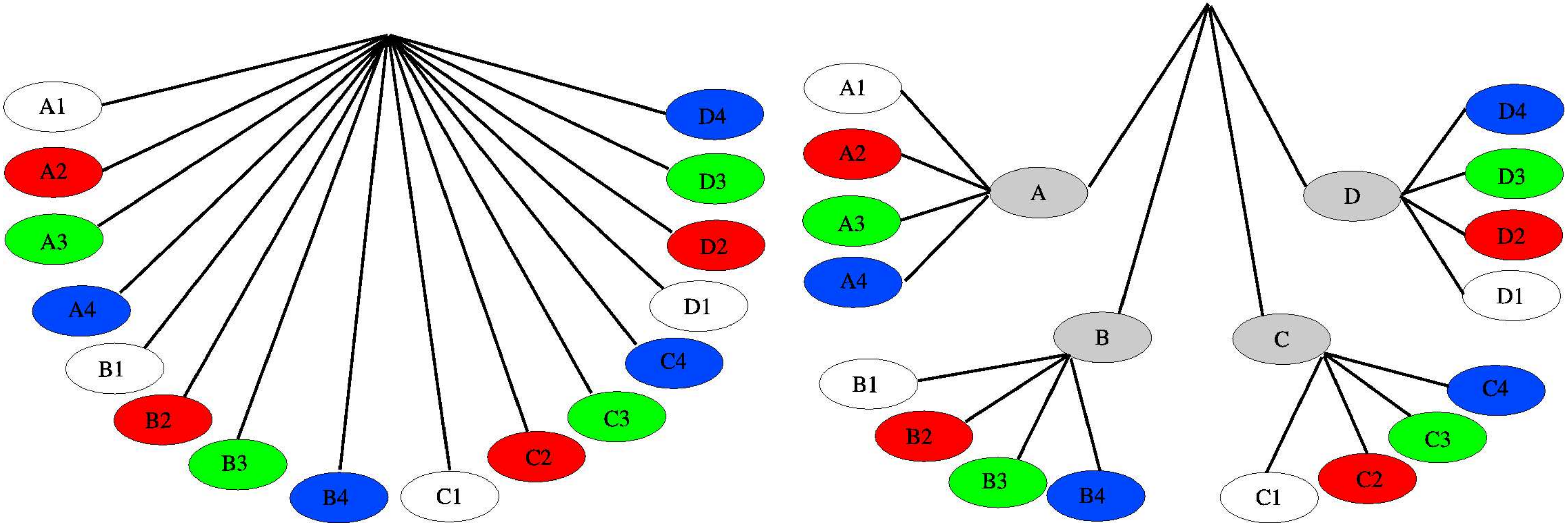
- Due to diversity of NDs small size of ensemble (around 20 members) will usually suffice.

- ENDs were shown to perform often better than other decomposition methods or multiclass-learners.

Hierarchically structured classes

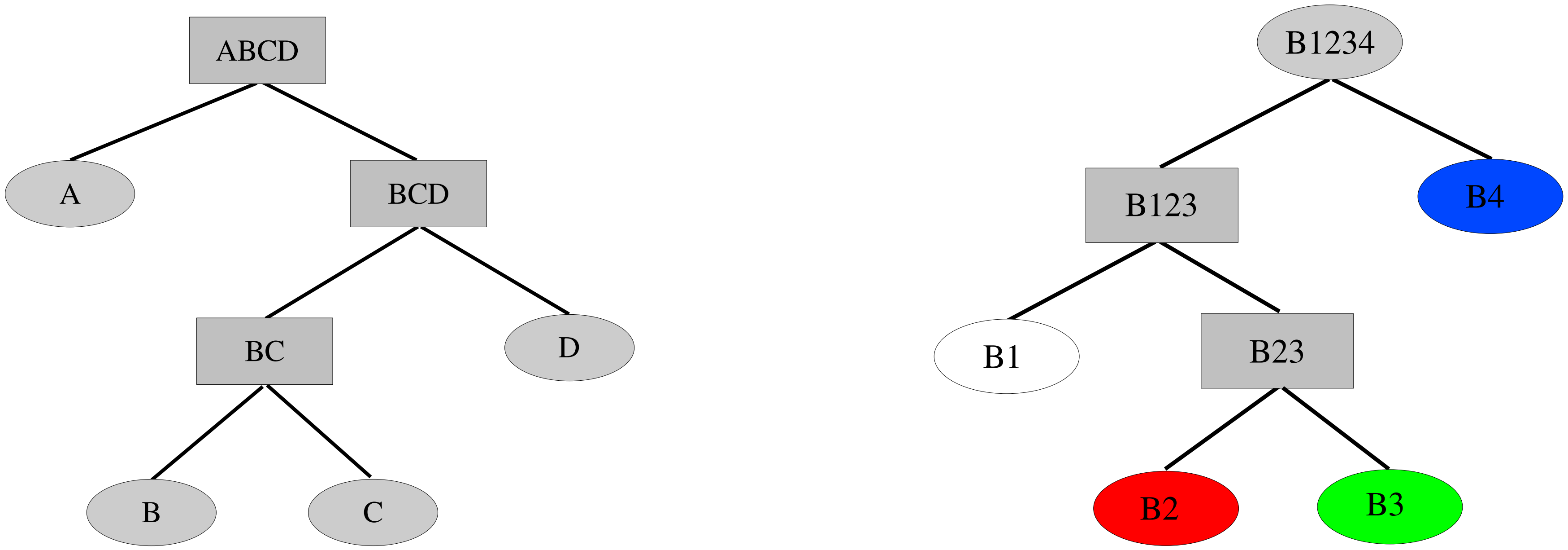


Simplifying a classification task by using a hierarchy



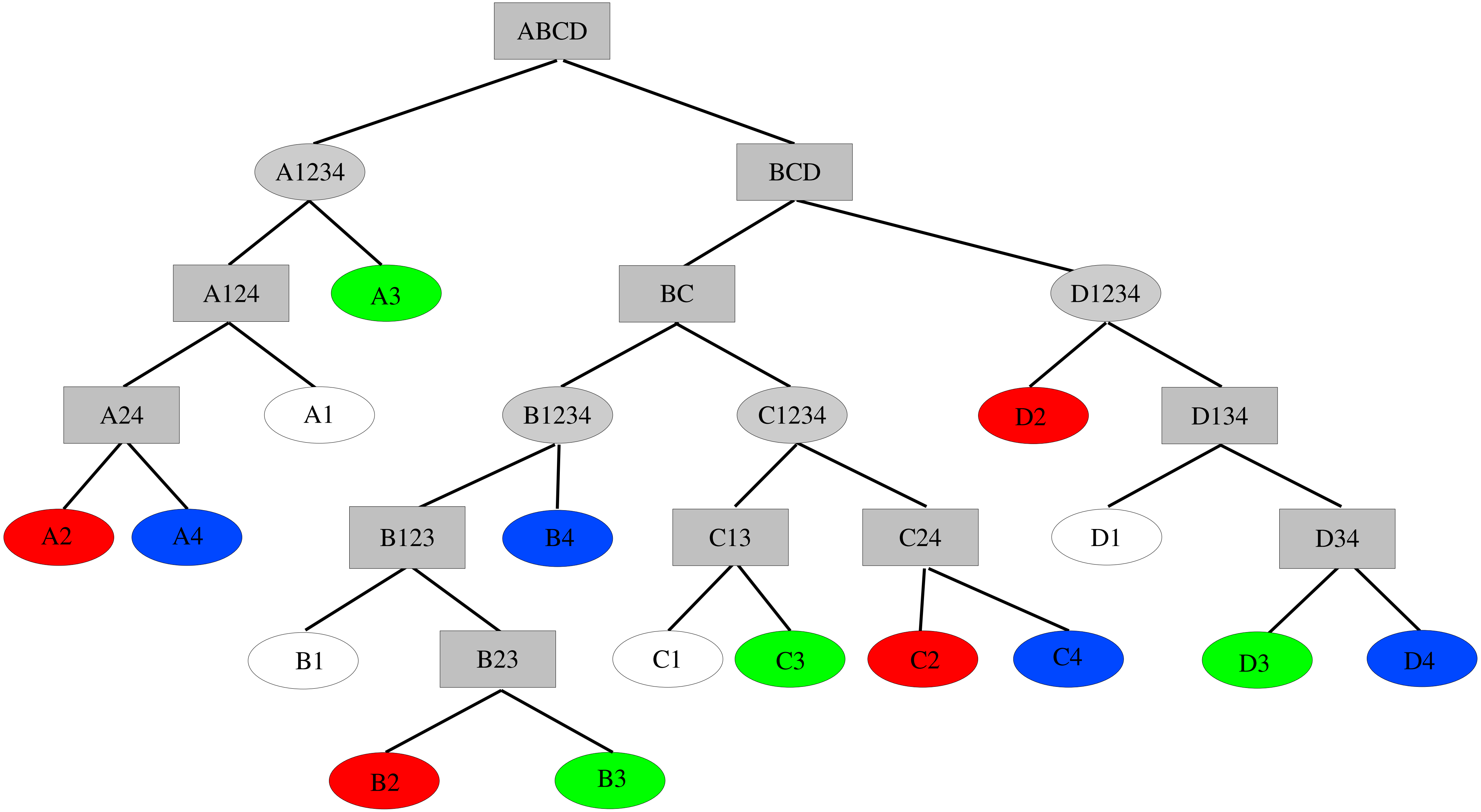
Creating hierarchically nested dichotomies

- Create an ND for the classification problem concerning the superclasses.
- For each superclass: create an ND for the classification problem concerning the subclasses of the respective superclass.



Each one out of $T(4) = 15$ binarizations.

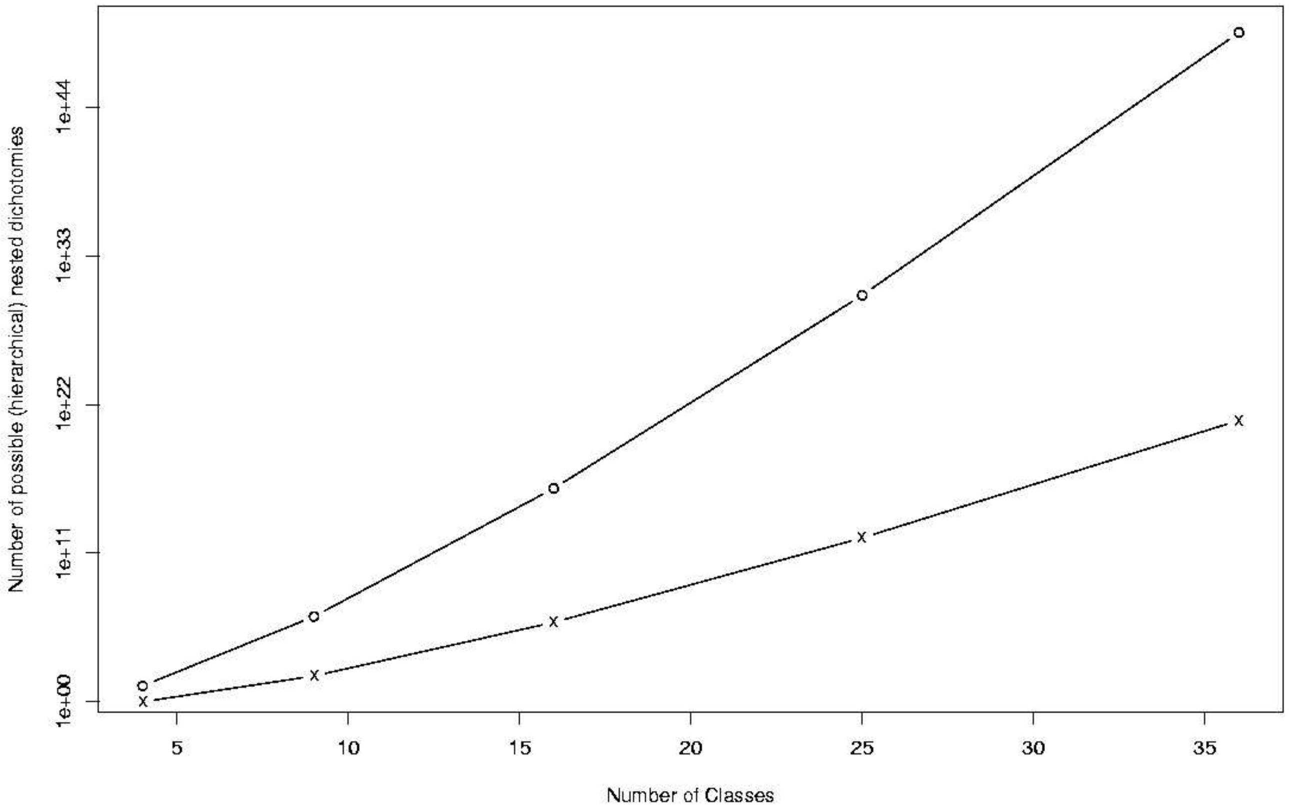
Resulting HND



One out of $T(4)^5 = 759,375$ possible HNDs.

Restriction of the space of possible NDs by a hierarchy

Assuming a completely balanced hierarchy describing n classes in l levels where at each level the respective superclass is divided in c subclasses ($n = c^l$):



$$\begin{aligned} T(n) &= (2n-3) \times T(n-1) \\ T(1) &= 1 \end{aligned}$$

$$\begin{aligned} H(n) &= H(c^l) \\ &= T(c)^{\frac{1-c^l}{1-c}} \end{aligned}$$

$$\begin{aligned} o &: T(c^2) \\ x &: H(c^2) \end{aligned}$$

Diversity of HNDs

- Space of HNDs is still growing over-exponentially.

- Single HND is still biased towards a more restrictive (binary) hierarchy.

- Building ensembles of HNDs can still be expected to reduce error variance.

Evaluation on example

Group	Method	Cross-validation			
		TPR	FPR	PPV	F1
A	SVM	0.5	0.0333	undef	undef
	END	0.775	0.015	0.8457	0.8088
	EHND	0.92	0.0053	0.92	0.92
B	SVM	0.75	0.0167	undef	undef
	END	0.84	0.0107	0.8433	0.8417
	EHND	0.93	0.0047	0.9294	0.9297
C	SVM	0.895	0.007	0.8946	0.8948
	END	0.76	0.016	0.7903	0.7749
	EHND	0.955	0.003	0.9553	0.9551
D	SVM	0.985	0.001	0.9851	0.985
	END	0.92	0.0053	0.9217	0.9208
	EHND	0.985	0.001	0.9851	0.985
Total	SVM	0.7825	0.0145	undef	undef
	END	0.8238	0.0117	0.8503	0.8368
	EHND	0.9475	0.0035	0.9474	0.9475

Application to Fold Recognition

Dataset Ding and Dubchak, comparison to machine learning approaches:

Approach	Prediction accuracy (Q)
Ding and Dubchak	
NN (OvO)	41.8 %
SVM (OvO)	45.2 %
SVM (uOvO)	51.1 %
SVM (AvA)	56.0 %
Chung et al.	
RBFN	49.4 %
Hierarchical Structure (MLP)	44.7 %
Hierarchical Structure (RBFN)	56.4 %
Hierarchical Structure (GRNN)	45.2 %
Hierarchical Structure (SVM)	53.8 %
Huang et al.	56.36 %
Chinnasamy et al.	58.18 %
ENDs	
(SVM)	58.96 %
(Bagged PART)	57.64 %
EHNDs	
four structural classes (SVM)	58.18 %
five structural classes (SVM)	58.44 %
five structural classes (Bagged PART)	58.7 %

Application to Fold Recognition

Dataset McGuffin as adapted by Bindewald et al., comparison to alignment based and machine learning methods, different evaluation procedures:

Approach	Prediction accuracy (Q)
PDB-BLAST	13.25 %
GenTHREADER	14 %
MANIFOLD	33.96 %
Alignment Combination	42 %
J48 *leave-one-out* – known-vs-complete*	21.43 %
EHNDs	
leave-one-out – known-vs-complete* (J48)	32.94 %
leave-one-out – known-vs-complete* (Bagged PART)	42.06 %
Bagged PART *10-fold cross-validation – known*	43.25 %
ENDs *10-fold cross-validation – known* (Bagged PART)	46.03 %
EHNDs	
10-fold cross-validation – known (Bagged PART)	45.63 %
20-fold cross-validation – known (Bagged PART)	45.24 %
leave-one-out – known (Bagged PART)	44.44 %

Conclusions

- EHNDs outperform ENDs on synthetic data exhibiting a pronounced hierarchical structure.

- Both, ENDs and EHNDs, improve considerably in comparison to their respective base learners (on synthetic and on fold recognition data).

- ENDs and EHNDs perform well in comparison to established methods on several fold recognition datasets. Some datasets, however, are more favorable to alignment based methods.

Conclusions

- Improvement of EHNDs w.r.t. ENDs can theoretically be expected if and only if
  1. the given hierarchy is reflecting the relations between classes according to an arbitrary similarity measure,

     and
  2. the respective similarity of classes is detectable in the features representing instances of classes.
- ENDs and EHNDs are close to one another in terms of accuracy on fold recognition datasets.
  - The hierarchies (SCOP and CATH) may not be well reflected in the established feature spaces.
  - Furthermore, SCOP and CATH differ considerably, so even the hierarchy may not be reflecting the relations between classes well.
- Preliminary idea: The tradeoff between ENDs and EHNDs might be helpful in evaluation of new feature spaces given a reliable hierarchy.