Building Ensembles of Adaptive Nested Dichotomies with Random-Pair Selection

Tim Leathart(✉), Bernhard Pfahringer, and Eibe Frank

Department of Computer Science, University of Waikato, Hamilton, New Zealand
tml15@students.waikato.ac.nz, {bernhard,eibe}@cs.waikato.ac.nz

Abstract. A system of nested dichotomies is a method of decomposing a multi-class problem into a collection of binary problems. Such a system recursively applies binary splits to divide the set of classes into two subsets, and trains a binary classifier for each split. Although ensembles of nested dichotomies with random structure have been shown to perform well in practice, using a more sophisticated class subset selection method can be used to improve classification accuracy. We investigate an approach to this problem called random-pair selection, and evaluate its effectiveness compared to other published methods of subset selection. We show that our method outperforms other methods in many cases when forming ensembles of nested dichotomies, and is at least on par in all other cases. The software related to this paper is available at https://svn.cms.waikato.ac.nz/svn/weka/trunk/packages/internal/ensemblesOfNestedDichotomies/.

1 Introduction

Multi-class classification problems – problems with more than two classes – are commonplace in real world scenarios. Some learning methods can handle multi-class problems inherently, *e.g.*, decision tree inducers, but others may require a different approach. Even techniques such as decision tree inducers may benefit from methods that decompose a multi-class problem in some manner. Typically, a collection of binary classifiers is trained and combined in some way to produce a multi-class classification. This process is called binarization. Popular techniques for adapting binary classifiers to multi-class problems include pairwise classification [11], one-vs-all classification [15], and error correcting output codes [5]. Ensembles of nested dichotomies [8] have been shown to be an effective substitute to these methods. Depending on the base classifier used, they can outperform both pairwise classification and error-correcting output codes [8].

In a nested dichotomy, the set of classes is split into two subsets recursively until there is only one class in each subset. Nested dichotomies are represented as binary tree structures (Fig. 1). At each node of a nested dichotomy, a binary classifier is learned to classify instances as belonging to one of the two subsets of classes. A nice feature of nested dichotomies is that class probability estimates can be computed in a natural way if the binary classifier used at each node can output two-class probability estimates.

P. Frasconi et al. (Eds.): ECML PKDD 2016, Part II, LNAI 9852, pp. 179–194, 2016.
DOI: 10.1007/978-3-319-46227-1_12

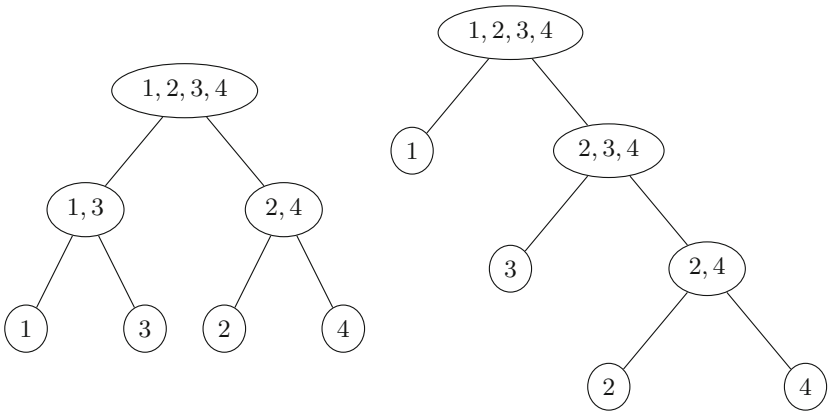


Fig. 1. Two examples of nested dichotomies for a four class problem.

The number of nested dichotomies for a c-class problem increases exponentially with the number of classes. One approach is to sample nested dichotomies at random to form an ensemble of them [8]. However, this may result in binary problems that are difficult to learn for the base classifier.

This paper is founded on the observation that some classes are generally easier to separate than others. For example, in a dataset of images of handwritten digits, the digits '5' and '6' are are much more difficult to distinguish than the digits '0' and '1'. This means that if '5' and '6' were put into opposite class subsets, the base classifier would have a more difficult task to discriminate the two subsets than if they were grouped together. Moreover, if the base classifier assigns high probability to an incorrect branch when classifying a test instance, it is unlikely that the final prediction will be correct. Therefore, we should try to group similar classes into the same class subsets whenever possible, and separate them in lower levels of the tree near the leaf nodes.

In this paper, we propose a method for semi-random class subset selection, which we call "random-pair selection", that attempts to group similar classes together for as long as possible. This means that the binary classifiers close to the root of the tree of classes can learn to distinguish higher-level features, while the ones close to the leaf nodes can focus on the more fine-grained details between similar classes. We evaluate this method against other published class subset selection strategies.

This paper is structured as follows. In Sect. 2, we give a review of other adaptations of ensembles of nested dichotomies. In Sect. 3, we describe the random-pair selection strategy and give an overview of how it works. We also cover theoretical advantages of our method over other methods, and give an analysis of how this strategy affects the space of possible nested dichotomy trees to sample from. In Sect. 4, we evaluate these methods and compare them to other class subset selection techniques.

2 Related Work

The original framework of ensembles of nested dichotomies by Frank and Kramer was proposed in 2004 [8]. In this framework, a binary tree is sampled randomly from the set of possible trees, based on the assumption that each nested dichotomy is equally likely to be useful *a priori*. By building an ensemble of nested dichotomies in this manner, Frank and Kramer achieved results that are competitive with other binarization techniques using decision trees and logistic regression as the two-class models for each node.

There have been a number of adaptations of ensembles of nested dichotomies since, mainly focusing on different class selection techniques. Dong *et al.* propose to restrict the space of nested dichotomies to only consist of structures with balanced splits [6]. Doing this regulates the depth of the trees, which can reduce the size of the training data for each binary classifier and thus has a positive effect on the runtime. It was shown empirically that this method has little effect on accuracy. Dong *et al.* also consider nested dichotomies where the number of instances per subset is approximately balanced at each split, instead of the number of classes. This also reduces the runtime, but can aversely effect the accuracy in rare cases.

The original framework of ensembles of nested dichotomies uses randomization to build an ensemble, *i.e.*, the structure of each nested dichotomy in the ensemble is randomly selected, but built from the same data. Rodriguez *et al.* explore the use of other ensemble techniques in conjunction with nested dichotomies [16]. The authors found that improvements in accuracy can be achieved by using bagging [3], AdaBoost [9] and MultiBoost [17] with random nested dichotomies as the base learner, compared to solely randomizing the structure of the nested dichotomies. The authors also experimented with different base classifiers for the nested dichotomies, and found that using ensembles of decision trees as base classifiers yielded favourable results compared to individual decision trees.

Duarte-Villaseñor *et al.* propose to split the classes more intelligently than randomly by using various clustering techniques [7]. They first compute the centroid of each class. Then, at each node of a nested dichotomy, they select the two classes with the furthest centroids as initial classes for each subset. Once the two classes have been picked, the remaining classes are assigned to one of the two subsets based on the distance of their centroids to the centroids of the initial classes. Duarte-Villaseñor *et al.* evaluate three different distance measures for determining the furthest centroids, taking into account the position of the centroids, the radius of the clusters and average distance of each instance from the centroid. They found that these class subset selection methods gave superior accuracy to the random methods previously proposed when the nested dichotomies were used for boosting.

3 Random-Pair Selection

We present a class selection strategy for choosing subsets in a nested dichotomy called random-pair selection. This has the same intention as the centroid-based methods proposed by Duarte-Villaseñor *et al.* [7]. Our method differs in that it takes a more direct approach to discovering similar classes by using the actual base classifier to decide which classes are more easily separable. Moreover, it incorporates an aspect of randomization.

3.1 The Algorithm

The process for constructing a nested dichotomy with random-pair selection is as follows:

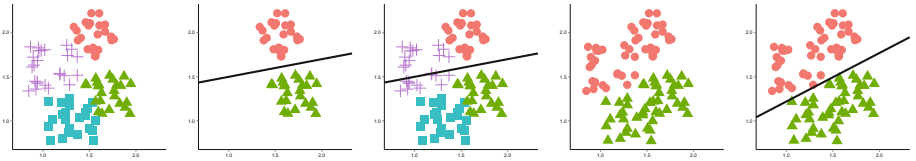
1. Create a root node for the tree.
2. If the class set C has only one class, then create a leaf node.
3. Otherwise, split C into two subsets by the following:
   (a) Select a pair of classes $c_1, c_2 \in C$ at random, where C is the set of all classes present at the current node.
   (b) Train a binary classifier using these two classes as training data. Then, use the remaining classes as test data, and observe which of the initial classes the majority of instances of each test class are classified as.[1]
   (c) Two subsets are created, using the initial classes: $s_1 = \{c_1\}, s_2 = \{c_2\}$
   (d) The test classes $c_n \in C \setminus \{c_1, c_2\}$ are added to s_1 or s_2 based on whether c_n is more likely to be classified as c_1 or c_2.
   (e) A new binary model is trained using the full data at the node, using the new class labels s_1 and s_2 for each instance.
4. Create new nodes for both s_1 and s_2 and recurse for each child node from Step 2.

This selection algorithm is illustrated in Fig. 2. The process for making predictions when using this class selection method is identical to the process for the original ensembles of nested dichotomies. Assuming that the base classifier can produce class probability estimates, the probability of an instance belonging to a class is the product of the estimates given by the binary classifiers on the path from the root to the leaf node corresponding to the particular class.

3.2 Analysis of the Space of Nested Dichotomies

To build an ensemble of nested dichotomies, a set of nested dichotomies needs to be sampled from the space of all nested dichotomies. The size of this space grows very quickly as the number of classes increases. Frank and Kramer calculate that the number of potential nested dichotomies is $(2c-3)!!$ for a c-class problem [8]. For a 10-class problem, this equates to $34,459,425$ distinct systems of nested

[1] When the dataset is large, it may be sensible to subsample the training data at each node when performing this step.



(a) (b) (c) (d) (e)

Fig. 2. Random-Pair Selection. (a) Original multi-class data. (b) Two classes are selected at random, and a binary classifier is trained on this data. (c) The binary classifier is tested on the other classes. The majority of the 'plus' class is classified as 'circle', and all of the 'square' class is classified as 'triangle'. (d) Combine the classes into subsets based on which of the original classes each new class is more likely to be classified as. (e) Learn another binary classifier, which will be used in the final nested dichotomy tree.

dichotomies. Using a class-balanced class-subset selection strategy reduces this number:

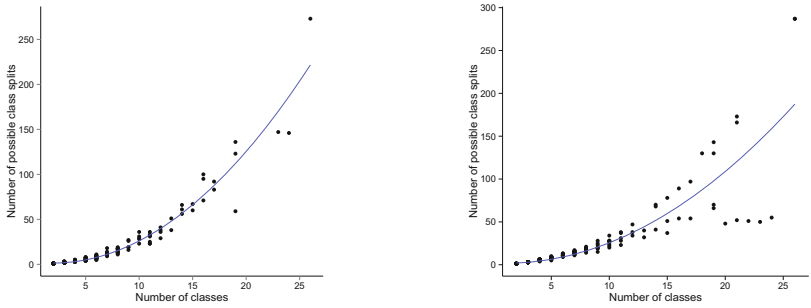
$$T(c) = \begin{cases} \frac{1}{2}\binom{c}{c/2}T(\frac{c}{2})T(\frac{c}{2}), & \text{if } c \text{ is even} \\ \binom{c}{(c+1)/2}T(\frac{c+1}{2})T(\frac{c-1}{2}), & \text{if } c \text{ is odd} \end{cases} \tag{1}$$

where $T(2) = T(1) = 1$ [6]. The number of class-balanced nested dichotomies is still very large, giving 113,400 possible nested dichotomies for a 10-class problem. The subset selection method based on clustering [7] takes this idea to the extreme, and gives only a single nested dichotomy for any given number of classes because the class subset selection is deterministic. Even though the system produced by this subset selection strategy is likely to be a useful one, it does not lend itself well to ensemble methods.

The size of the space of nested dichotomies that we sample using the random-pair selection method varies for each dataset, and is dependent on the base classifier. The upper bound for the number of possible binary problems at each node is the number of ways to select two classes at random from a c-class dataset, *i.e.*, $\binom{c}{2}$. In practice, many of these randomly chosen pairs are likely to produce the same class subsets under our method, so the number of possible class splits is likely to be lower than this value. For illustrative purposes, we empirically estimate this value for the logistic regression base learner. We enumerate and count the number of possible class splits for our splitting method at each node of a nested dichotomy for a number of datasets, and plot this number against the number of classes at the corresponding node (Fig. 3a). We also show a similar plot for the case where C4.5 is used as the base classifier (Fig. 3b). Fitting a second degree polynomial to the data for logistic regression yields

$$p(c) = 0.3812c^2 - 1.4979c + 2.9027. \tag{2}$$

Assuming we apply logistic regression, we can estimate the number of possible class splits for an arbitrary number of classes based on this expression by making a rough estimate of the distribution of classes at each node. Nested dichotomies



(a) Logistic regression (b) C4.5

Fig. 3. Number of possible splits under a random-pair selection method vs number of classes for a number of UCI datasets.

Table 1. The number of possible nested dichotomies for up to 12 classes for each class subset selection technique. The first two columns are taken from [6], and the random-pair column is estimated from (3).

Number of classes	Number of nested dichotomies	Number of class-balanced nested dichotomies	Number of random-pair nested dichotomies
2	1	1	1
3	3	3	1
4	15	3	5
5	105	30	15
6	945	90	36
7	10,395	315	182
8	135,135	315	470
9	2,027,025	11,340	1,254
10	34,459,425	113,400	7,002
11	654,729,075	1,247,400	28,189
12	13,749,310,575	3,742,200	81,451

constructed with random-pair selection are not guaranteed to be balanced, so we average the class subset proportions over a large sample of nested dichotomies on different datasets to find that the two class subsets contain $\frac{1}{3}$ and $\frac{2}{3}$ respectively of the classes on average. Given this information, we can estimate the number of possible nested dichotomies with logistic regression by the recurrence relation

$$T(c) = p(c)T(\frac{c}{3})T(\frac{2c}{3}) \tag{3}$$

where $T(c) = 1$ when $c \leq 2$. Table 1 shows the number of distinct nested dichotomies that can be created for up to 12 classes for the random-pair selection

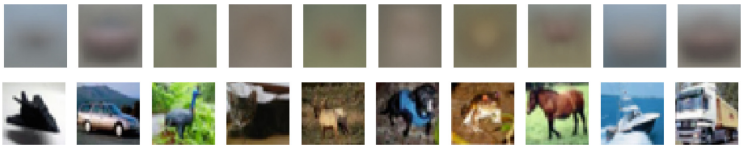


Fig. 4. Class centroids of the training component of the CIFAR-10 dataset (above). Samples from each class (below).

method, class-balanced and completely random selection when we apply this estimate.

3.3 Advantages Over Centroid Methods

Random-pair selection has two theoretical advantages compared to the centroid-based methods proposed by the authors of [7]: (a) an element of randomness makes it more suitable for ensemble learning, and (b) it adapts to the base classifier that is used.

In the centroid-based methods, each class split is deterministically chosen based on some distance metric. This means that the structure of every nested dichotomy in an ensemble will be the same. This is less important in ensemble techniques that alter the dataset or weights inside the dataset (*e.g.*, bagging or boosting). However, an additional element of randomization in ensembles is typically beneficial. When random-pair selection is employed, the two initial classes are randomly selected in all nested dichotomies, increasing the total number of nested dichotomies that can be constructed as discussed in the previous section.

Centroid-based methods assume that a smaller distance between two class centroids is indicative of class similarity. While it is true that this is often the case, sometimes the centroids can be relatively meaningless. An example is the CIFAR-10 dataset, a collection of small natural images of various categories such as cats, dogs and trucks [12]. The classes are naturally divided into two subsets – animals and vehicles. Figure 4 shows an image representation of the centroids of each class, and a sample image from the respective class below it. It is clear to see that most of these class centroids do not contain much useful information for discriminating between the classes.

This effect is clearer when evaluating a simple classifier that classifies instances according to the closest centroid of the training data. For illustrative purposes, see the confusion matrix of such a classifier when trained on the CIFAR-10 dataset (Fig. 5). It is clear to see from the confusion matrix that the centroids cannot be relied upon to produce meaningful predictions in all cases for this data.

A disadvantage of random-pair selection compared to centroid-based methods is an increase in runtime. Under our method, we need to train additional base classifiers during the class subset selection process. However, the extra base classifiers are only trained on a subset of the data at a node, *i.e.*, only two of the

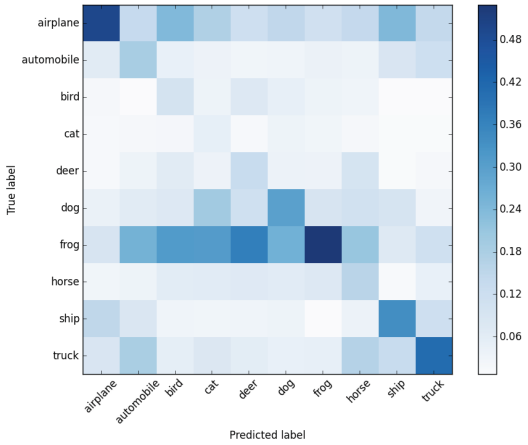


Fig. 5. Confusion matrix of a centroid classifier for the CIFAR-10 dataset. The darkness of each square corresponds with the number of instances classified as a particular class.

classes, and we can subsample this data during this step if we need to improve the runtime further.

4 Experimental Results

We present an evaluation of the random-pair selection method on 18 datasets from the UCI repository [13]. Table 2 lists and describes the datasets we used. We specifically selected datasets with at least five classes, as our method will not have a large impact on datasets with few classes. This is due to the fact that there is a relatively small number of possible nested dichotomies for small numbers of classes.

4.1 Experimental Setup

All experiments were conducted in WEKA [10], and performed with 10 times 10-fold cross validation.[2] The default settings in WEKA for the base learners and ensemble methods were used in our evaluation. We compared our class subset selection method (RPND) to nested dichotomies based on clustering (NDBC) [7], class-balanced nested dichotomies (CBND) [6], and completely random selection (ND) [8]. We did not compare against other variants of nested dichotomies such as data-balanced nested dichotomies [6], nested dichotomies based on clustering with radius [7] and nested dichotomies based on clustering with average radius [7], because they were found to either have the same or worse performance on average in [6] and [7] respectively. We used logistic regression and

[2] Our implementations can be found in the `ensemblesOfNestedDichotomies` package in WEKA.

Table 2. The datasets used in this evaluation

Dataset	Classes	Instances	Attributes	Dataset	Classes	Instances	Attributes
audiology	24	226	70	optdigits	10	5620	65
krkopt	18	28056	7	page-blocks	5	5473	11
LED24	10	5000	25	pendigits	10	10992	17
letter	26	20000	17	segment	7	2310	20
mfeat-factors	10	2000	217	shuttle	7	58000	10
mfeat-fourier	10	2000	77	usps	10	9298	257
mfeat-karhunen	10	2000	65	vowel	11	990	14
mfeat-morphological	10	2000	7	yeast	10	1484	9
mfeat-pixel	10	2000	241	zoo	7	101	18

C4.5 as the base learners for our experiments, as they occupy both ends of the bias-variance spectrum. In our results tables, a bullet (•) indicates a statistically significant accuracy gain, and an open circle (○) indicates a statistically significant accuracy reduction ($p = 0.05$) by using the random-pair method compared with another method. To establish significance, we used the corrected resampled paired t-test [14].

4.2 Single Nested Dichotomy

We expect that intelligent class subset selection will have a larger impact in small ensembles of nested dichotomies. This is due to the fact as ensembles grow larger, the worse performing ensemble members will not have as great an influence over the final predictions. Therefore, we first compare a single nested dichotomy using random-pair selection to a single nested dichotomy obtained with other class selection methods.

Table 3 shows the classification accuracy and standard deviations of each method when training a single nested dichotomy. When logistic regression is used as the base learner, compared to random methods (CBND and ND), we obtain a significant accuracy gain in most cases, and comparable accuracy in all others. When using C4.5 as the base learner, our method is preferable to random methods in some cases, with all other datasets showing a comparable accuracy.

In comparison to NDBC, gives similar accuracy overall, with three significantly better results, four significantly worse results, and the rest comparable over both base learners. It is to be expected that NDBC sometimes has better performance than our method when only a single nested dichotomy is built. This is because NDBC deterministically selects the class split that is likely to be the most easily separable. Our method attempts to produce an easily separable class subset selection from a pool of possible options, where each option is as likely as any other.

4.3 Ensembles of Nested Dichotomies

Ensembles of nested dichotomies typically outperform single nested dichotomies. The original method for creating an ensemble of nested dichotomies is a randomization approach, but it was later found that better performance can be obtained by bagging and boosting nested dichotomies [16]. For this reason, we consider three types of ensembles of nested dichotomies in our experiments: bagged, boosted with AdaBoost and boosted with MultiBoost (the latter two applied with resampling based on instance weights). We built ensembles of 10 nested dichotomies for these experiments.

Bagging. Table 4 shows the results of using bagging to construct an ensemble of nested dichotomies for each method and for both base learners. When logistic

Table 3. Accuracy of a single nested dichotomy with (a) logistic regression and (b) C4.5 as the base learner.

(a)

Dataset	RPND	NDBC	CBND	ND
audiology	75.36±8.45	72.47±8.80	68.55±9.61	71.91±9.85
krkopt	33.13±0.97	33.23±0.80	28.55±1.50 •	28.70±1.56 •
LED24	72.85±2.03	72.73±2.06	67.11±4.08 •	70.26±3.28 •
letter	67.70±2.72	72.23±0.93 ◦	47.98±3.08 •	53.10±4.36 •
mfeat-factors	95.04±1.99	96.62±1.19 ◦	91.83±2.20 •	93.08±2.15 •
mfeat-fourier	76.37±3.22	75.17±2.76	73.17±3.34 •	74.00±3.34
mfeat-karhunen	89.83±2.32	90.83±1.75	84.96±3.75 •	86.53±3.06 •
mfeat-morphological	72.64±3.25	70.45±3.03 •	62.31±7.79 •	66.40±5.19 •
mfeat-pixel	71.16±9.98	88.67±2.51 ◦	61.25±9.25	47.44±9.15 •
optdigits	92.72±2.06	92.00±1.10	87.83±3.01 •	90.95±2.60
page-blocks	96.17±0.75	95.77±0.77	95.44±0.84 •	95.61±0.86
pendigits	90.20±2.32	87.97±0.96 •	82.23±4.42 •	87.08±4.22
segment	94.02±2.40	88.76±1.91 •	87.36±4.16 •	89.11±3.93 •
shuttle	96.87±0.46	96.86±0.20	92.14±6.86	91.72±7.03 •
usps	87.47±1.47	87.64±1.06	84.70±2.26 •	85.83±1.97 •
vowel	81.80±4.46	80.83±4.10	47.86±8.67 •	53.08±8.98 •
yeast	58.35±3.89	59.00±3.58	56.43±4.20	55.91±3.90 •
zoo	90.41±9.15	87.55±9.32	88.88±9.34	89.00±8.65

(b)

Dataset	RPND	NDBC	CBND	ND
audiology	76.86±7.23	75.49±7.29	74.45±8.04	73.79±7.62
krkopt	70.04±2.45	69.33±0.99	64.83±1.78 •	65.13±2.19 •
LED24	72.68±2.12	72.99±1.72	72.07±2.08	72.22±2.05
letter	86.32±0.85	86.50±0.88	85.38±0.88 •	86.03±0.88
mfeat-factors	88.47±2.59	88.77±1.73	86.76±2.43	87.47±2.23
mfeat-fourier	74.46±3.09	73.97±2.90	72.63±2.97	73.03±3.29
mfeat-karhunen	82.04±2.84	82.56±2.66	80.11±3.15	80.18±3.28
mfeat-morphological	72.44±2.73	72.27±2.48	71.90±2.40	71.85±2.52
mfeat-pixel	81.83±3.23	81.36±2.79	77.13±3.61 •	79.44±3.91
optdigits	90.72±1.43	90.76±1.15	89.27±1.52 •	89.93±1.44
page-blocks	97.07±0.72	97.05±0.66	97.00±0.67	97.05±0.65
pendigits	95.92±0.70	95.81±0.62	95.60±0.67	95.79±0.68
segment	96.10±1.38	96.59±1.25	95.88±1.49	95.88±1.37
shuttle	99.97±0.02	99.98±0.02	99.97±0.02	99.97±0.03
usps	87.95±1.18	89.44±0.91 ◦	86.06±1.52 •	86.48±1.37 •
vowel	79.04±4.22	76.96±4.45	76.07±4.75	75.54±4.87
yeast	57.22±3.31	57.58±3.69	56.18±3.43	56.64±3.36
zoo	91.63±8.06	88.65±8.30	90.72±7.12	90.67±8.72

Table 4. Accuracy of an ensemble of 10 bagged nested dichotomies with (a) logistic regression and (b) C4.5 as the base learner.

(a)

Dataset	RPND	NDBC	CBND	ND
audiology	81.79±7.56	81.25±7.25	80.32±7.69	82.35±7.57
krkopt	33.77±0.78	33.29±0.77 •	31.73±0.98 •	31.99±0.94 •
LED24	73.56±1.90	73.42±2.01	73.50±1.94	73.49±1.85
letter	78.65±0.94	76.16±0.96 •	73.76±1.24 •	74.51±1.27 •
mfeat-factors	98.11±1.02	97.39±1.10 •	97.72±1.09	97.94±1.01
mfeat-fourier	83.08±2.18	80.03±2.25 •	82.16±2.66	82.14±2.39
mfeat-karhunen	95.66±1.54	93.67±1.75 •	94.88±1.56	94.89±1.57
mfeat-morphological	73.71±2.79	72.33±2.87	73.19±2.94	73.55±2.45
mfeat-pixel	94.70±1.95	93.15±1.49 •	90.96±2.51 •	83.65±4.01 •
optdigits	97.15±0.68	93.56±0.93 •	96.50±0.83 •	96.83±0.68
page-blocks	96.46±0.68	96.14±0.66 •	95.92±0.72 •	96.11±0.68 •
pendigits	95.93±0.80	88.90±1.08 •	94.61±1.00 •	95.12±0.88 •
segment	95.37±1.61	89.26±1.95 •	94.03±1.96 •	94.15±1.73 •
shuttle	96.74±0.24	96.86±0.21	94.94±1.52 •	94.86±1.39 •
usps	93.83±0.69	92.02±0.91 •	93.59±0.70	93.32±0.73 •
vowel	89.76±3.04	85.72±3.49 •	77.52±4.90 •	78.30±4.61 •
yeast	58.86±3.85	59.18±3.84	58.91±3.64	58.92±3.62
zoo	94.87±6.03	91.62±8.33	93.36±7.16	93.20±7.37

(b)

Dataset	RPND	NDBC	CBND	ND
audiology	79.76±7.32	80.33±6.11	80.65±7.29	79.30±7.30
krkopt	75.70±0.95	73.93±0.90 •	74.20±1.00 •	74.82±1.00 •
LED24	73.22±1.92	73.12±1.82	73.10±1.90	73.23±1.92
letter	93.81±0.55	92.73±0.66 •	93.92±0.50	94.07±0.49
mfeat-factors	95.27±1.58	93.37±1.76 •	95.80±1.40	95.44±1.52
mfeat-fourier	81.36±2.81	78.79±2.64 •	81.30±2.83	80.94±2.76
mfeat-karhunen	92.83±1.96	90.27±2.11 •	92.98±1.42	93.13±1.67
mfeat-morphological	73.38±2.61	72.78±2.72	73.07±2.83	73.37±2.62
mfeat-pixel	92.56±1.91	87.01±2.47 •	92.24±1.82	92.65±1.79
optdigits	97.09±0.70	95.34±0.90 •	97.04±0.72	97.00±0.72
page-blocks	97.41±0.64	97.29±0.62	97.39±0.59	97.36±0.63
pendigits	98.53±0.40	97.67±0.46 •	98.68±0.35	98.64±0.38
segment	97.45±1.09	97.52±1.11	97.54±1.14	97.53±0.88
shuttle	99.98±0.02	99.97±0.02	99.98±0.02	99.98±0.02
usps	94.63±0.59	93.85±0.72 •	94.52±0.59	94.61±0.70
vowel	87.69±3.52	85.82±3.73	89.15±3.46	88.26±3.25
yeast	59.86±3.29	59.55±3.38	59.93±3.54	59.72±3.79
zoo	93.81±7.17	91.70±7.77	93.57±6.81	94.36±6.17

regression is used as a base learner, our method outperforms all other methods in many cases. When C4.5 is used as a base learner, our method compares favourably with NDBC and achieves comparable accuracy to the random methods. Our method is better in a bagging scenario than NDBC because of the first problem highlighted in Sect. 3.3, *i.e.*, using the furthest centroids to select a class split results in a deterministic class split. Evidently, with bagged datasets, this method of class subset selection is too stable to be utilized effectively. Our method, on the other hand, is sufficiently unstable to be useful in a bagged ensemble.

AdaBoost. Table 5 shows the results of using AdaBoost to build an ensemble of nested dichotomies for each method and for both base learners. When comparing with the random methods, we observe a similar result to the bagged ensembles. When using logistic regression, we see a significant improvement in accuracy in

Table 5. Accuracy of an ensemble of 10 nested dichotomies boosted with AdaBoost with (a) logistic regression and (b) C4.5 as the base learner.

(a)

Dataset	RPND	NDBC	CBND	ND
audiology	81.42±7.38	80.31±6.92	79.87± 7.49	80.78± 7.50
krkopt	32.99±1.01	32.81±0.77	28.24± 1.47 •	28.66± 1.44 •
LED24	72.41±2.16	72.93±1.99	69.17± 2.77 •	70.44± 2.72 •
letter	71.39±2.50	71.44±1.49	47.42± 3.29 •	55.16± 5.35 •
mfeat-factors	97.71±1.09	97.66±0.99	97.11± 1.25	97.52± 1.17
mfeat-fourier	81.01±2.28	79.96±2.52	80.12± 2.43	80.13± 2.64
mfeat-karhunen	94.93±1.50	94.42±1.61	93.76± 1.54 •	94.01± 1.54
mfeat-morphological	72.81±2.82	71.02±3.10	66.73± 6.80 •	69.38± 5.53
mfeat-pixel	94.15±1.81	93.87±1.59	91.16± 2.39 •	86.21± 3.48 •
nursery	92.51±0.70	92.52±0.70	92.29± 0.74	92.38± 0.69
optdigits	97.01±0.69	96.84±0.77	96.26± 0.74 •	96.37± 0.86 •
page-blocks	96.09±0.80	95.93±0.75	95.43± 0.84	95.77± 0.90
pendigits	94.94±0.93	94.83±0.77	93.86± 1.30	93.67± 1.03 •
segment	94.94±1.40	94.66±1.48	93.88± 1.93	93.82± 1.84
shuttle	96.83±0.45	96.86±0.26	96.51± 1.57	96.40± 2.18
usps	92.03±0.88	91.83±0.86	91.91± 0.91	91.66± 0.85
vowel	90.59±3.11	89.74±3.10	48.45±10.68 •	58.93±11.42 •
yeast	57.97±3.78	58.39±3.62	56.90± 4.05	56.56± 3.66
zoo	94.95±6.40	94.96±6.33	94.38± 7.44	94.77± 6.19

(b)

Dataset	RPND	NDBC	CBND	ND
audiology	83.64±7.37	83.29±6.68	82.63±6.87	82.58±7.36
krkopt	81.01±0.78	79.37±0.80 •	77.25±0.95 •	78.36±1.04 •
LED24	69.59±2.13	69.49±2.11	69.04±1.95	69.42±1.78
letter	94.58±0.49	94.37±0.48	94.30±0.49	94.60±0.55
mfeat-factors	95.75±1.36	95.31±1.48	95.49±1.38	95.62±1.37
mfeat-fourier	80.43±2.74	79.54±2.60	80.12±2.49	80.74±2.47
mfeat-karhunen	93.20±1.80	92.67±1.83	92.96±1.76	92.85±1.84
mfeat-morphological	70.48±3.10	70.45±3.19	70.13±2.84	70.50±2.45
mfeat-pixel	93.76±1.53	93.27±1.80	92.48±1.80 •	93.01±1.83
optdigits	97.31±0.72	97.23±0.70	97.25±0.68	97.20±0.70
page-blocks	97.05±0.62	97.05±0.66	97.11±0.64	97.11±0.66
pendigits	98.95±0.30	98.89±0.33	98.91±0.30	98.93±0.28
segment	98.23±0.84	98.24±0.84	98.09±0.86	98.09±0.94
shuttle	99.99±0.01	99.99±0.01	99.99±0.01	99.99±0.01
usps	94.85±0.64	94.86±0.64	94.41±0.72	94.59±0.66
vowel	91.95±2.71	90.73±3.00	91.28±2.82	91.30±2.78
yeast	57.39±3.76	57.42±4.02	56.93±3.27	57.25±4.19
zoo	95.45±6.19	95.53±6.39	95.15±6.21	95.36±6.13

many cases, and when C4.5 is used, we typically see comparable results, with a small number of significant accuracy gains. When comparing with NDBC, we see a small improvement for the vast majority of datasets, but these differences are almost never individually significant. In one instance (krkopt with C4.5 as the base learner), we achieve a significant accuracy gain using our method.

MultiBoost. Table 6 shows the results of using MultiBoost to build an ensemble of nested dichotomies for each method and for both base learners. Compared to the random methods, again we see similar results to the other ensemble methods – using logistic regression as the base learner results in many significant improvements, and using C4.5 as the base learner typically produces comparable results, with few significant improvements. In comparison to NDBC,

Table 6. Accuracy of an ensemble of 10 nested dichotomies boosted with MultiBoost with (a) logistic regression and (b) C4.5 as the base learner.

(a)

Dataset	RPND	NDBC	CBND	ND
audiology	80.55±7.80	80.05±7.20	78.90± 7.51	79.53± 7.73
krkopt	32.99±1.01	32.81±0.77	28.24± 1.47 •	28.66± 1.44 •
LED24	73.38±1.81	73.31±2.15	72.01± 2.67	72.75± 2.38
letter	77.29±1.83	75.36±1.03 •	47.42± 3.29 •	55.85± 6.25 •
mfeat-factors	97.82±1.16	97.70±1.09	97.40± 1.31	97.53± 1.17
mfeat-fourier	82.12±2.28	80.22±2.28 •	80.22± 2.35 •	80.72± 2.44
mfeat-karhunen	95.22±1.59	94.70±1.57	93.94± 1.62 •	94.17± 1.71
mfeat-morphological	73.63±2.80	72.33±2.64	67.52± 7.04 •	70.40± 5.74
mfeat-pixel	94.37±1.48	94.16±1.30	91.89± 2.71 •	86.37± 4.74 •
optdigits	97.03±0.57	96.10±0.79 •	96.26± 0.78 •	96.47± 0.83 •
page-blocks	96.39±0.69	96.10±0.72	96.01± 0.68 •	96.19± 0.74
pendigits	96.02±0.73	94.27±1.32 •	94.17± 1.05 •	94.76± 0.92 •
segment	95.56±1.40	94.11±1.92 •	94.12± 1.93 •	94.35± 1.63 •
shuttle	96.89±0.27	96.87±0.24	96.63± 1.53	96.65± 1.59
usps	93.12±0.78	92.45±0.84 •	92.62± 0.83	92.57± 0.84
vowel	89.53±3.15	87.52±3.03	48.92±11.26 •	60.91±12.38 •
yeast	58.28±4.19	58.60±3.93	57.13± 4.03	57.03± 3.88
zoo	94.97±6.49	94.65±6.79	94.46± 7.35	94.07± 7.02

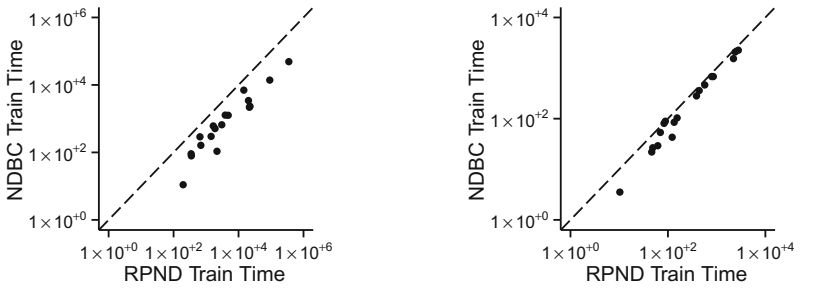
(b)

Dataset	RPND	NDBC	CBND	ND
audiology	81.32±7.06	82.14±7.39	81.25±7.48	80.32±7.37
krkopt	76.22±0.80	75.05±0.84 •	73.54±1.03 •	74.58±1.14 •
LED24	72.27±2.00	71.90±1.99	71.78±1.89	71.96±1.99
letter	93.98±0.47	93.65±0.53	93.78±0.55	93.98±0.46
mfeat-factors	95.63±1.33	94.82±1.45	95.32±1.46	95.14±1.48
mfeat-fourier	80.46±2.40	79.54±2.36	80.36±2.57	80.68±3.00
mfeat-karhunen	92.88±1.95	91.82±1.91	92.16±2.03	92.64±1.81
mfeat-morphological	71.30±2.75	71.26±2.85	71.32±3.11	71.75±2.84
mfeat-pixel	93.10±1.71	91.15±1.86 •	91.75±1.67 •	92.40±1.90
optdigits	97.00±0.70	96.80±0.75	96.91±0.73	97.00±0.69
page-blocks	97.33±0.65	97.24±0.63	97.34±0.64	97.29±0.66
pendigits	98.78±0.35	98.69±0.35	98.78±0.33	98.75±0.28
segment	97.90±0.93	98.06±0.94	97.79±0.95	97.88±0.99
shuttle	99.99±0.02	99.99±0.02	99.99±0.02	99.99±0.01
usps	94.67±0.65	94.48±0.64	94.25±0.58	94.33±0.71
vowel	88.60±3.40	88.33±3.61	88.79±3.18	88.34±3.56
yeast	58.91±3.58	58.91±3.56	58.53±3.63	58.35±3.92
zoo	95.09±6.73	94.17±7.34	94.26±6.48	95.66±6.11

we see many small (although statistically insignificant) improvements across both base learners, with some significant gains in accuracy on some datasets.

4.4 Training Time

Figure 6 shows the training time in milliseconds for training a single RPND and a single NDBC, with logistic regression and C4.5 as the base learners for each of the datasets used in this evaluation. As can be seen from the plots, there is a computational cost for building an RPND over an NDBC, which is to be expected as there is an additional classifier trained and tested at each split node of the tree. The gradient of both plots is approximately one, which indicates that our method does not add additional computational complexity to the problem. The runtime is comparatively worse for logistic regression than for C4.5.



(a) Logistic regression (b) C4.5

Fig. 6. Log-log plots of the training time for a single RPND and a single NDBC, for both base learners.

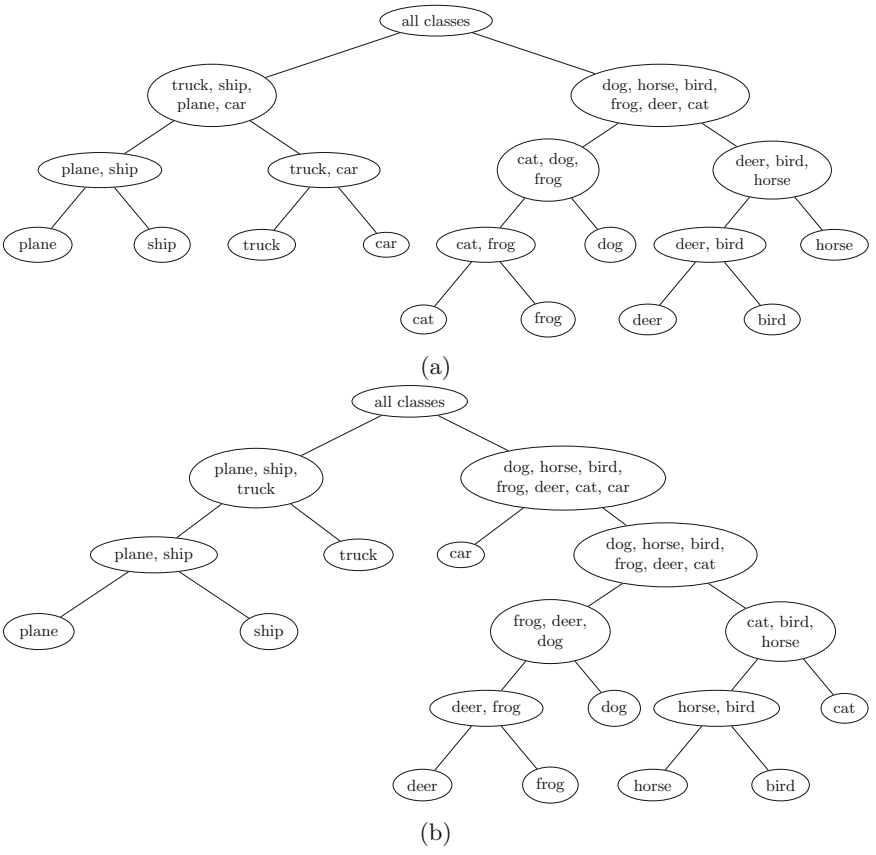


Fig. 7. Nested dichotomies trained on CIFAR-10, with (a) random-pair selection, and (b) centroid-based selection.

4.5 Case Study: CIFAR-10

To test how well our method adapts to other base learners, we trained nested dichotomies with convolutional networks as the base learners to classify the CIFAR-10 dataset [12]. Convolutional networks learn features from the data automatically, and perform well on high dimensional, highly correlated data such as images. We implemented the nested dichotomies and convolutional networks in Python using Lasagne [4], a wrapper for Theano [1,2]. The convolutional network that we used as the base learner is relatively simple; it has two convolutional layers with 32 5×5 filters each, one 3×3 maxpool layer with 2×2 stride after each convolutional layer, and one fully-connected layer of 128 units before a softmax layer.

As discussed in Sect. 3.3, the centroids for a dataset like CIFAR-10 appear to not be very descriptive, and as such, we expect NDBC with convolutional networks as the base learner to produce class splits that are not as well founded as those in RPND. We present a visualisation of the NDBC produced from the CIFAR-10 dataset, and an example of a nested dichotomy built with random-pair selection (Fig. 7). We can see that both methods produce a reasonable dichotomy structure, but there are some cases in which the random-pair method results in more intuitive splits. For example, the root node of the RPND splits the full set of classes into the two natural subsets (vehicles and animals), whereas the NDBC omits the 'car' class from the left-hand subset. Two pairs of similar classes in the animal subset – 'deer' and 'horse', and 'cat' and 'dog' – are kept together until near the leaves in the RPND, but are split up relatively early in the NDBC. Despite this, the accuracy and runtime of both methods were comparable. Of course, the quality of the nested dichotomy under random-pair selection is dependent on the initial pair of classes that is selected. If two classes that are similar to each other are selected to be the initial random pair, the tree can end up with splits that make less intuitive sense.

5 Conclusion

In this paper, we have proposed a semi-random method of class subset selection in ensembles of nested dichotomies, where the class selection is directly based on the ability of the base classifier to separate classes. Our method non-deterministically produces an easily separable class-split, which not only improves the accuracy over random methods for a single classifier, but also for ensembles of nested dichotomies. Our method also outperforms other non-random methods when nested dichotomies are used in a bagged ensemble and an ensemble boosted with MultiBoost, and otherwise gives comparable results.

In the future, it would be interesting to explore selecting several random pairs of classes at each node, and choosing the best of the pairs to create the final class subsets. This will obviously increase the runtime, but may help to produce more accurate individual classifiers and small ensembles. We also wish to explore the use of convolutional networks in nested dichotomies further.

Acknowledgements. This research was supported by the Marsden Fund Council from Government funding, administered by the Royal Society of New Zealand. The authors also thank NVIDIA for donating a K40c GPU to support this research.

References

1. Bastien, F., Lamblin, P., Pascanu, R., Bergstra, J., Goodfellow, I.J., Bergeron, A., Bouchard, N., Bengio, Y.: Theano: new features and speed improvements. In: Deep Learning and Unsupervised Feature Learning NIPS 2012 Workshop (2012)
2. Bergstra, J., Breuleux, O., Bastien, F., Lamblin, P., Pascanu, R., Desjardins, G., Turian, J., Warde-Farley, D., Bengio, Y.: Theano: a CPU and GPU math expression compiler. In: Proceedings of the Python for Scientific Computing Conference (SciPy), June 2010. Oral Presentation
3. Breiman, L.: Bagging predictors. Mach. Learn. **24**(2), 123–140 (1996)
4. Dieleman, S., Schlter, J., Raffel, C., Olson, E., Snderby, S.K., Nouri, D., Maturana, D., Thoma, M., Battenberg, E., Kelly, J., Fauw, J.D., Heilman, M., diogo149, McFee, B., Weideman, H., takacsg84, peterderivaz, Jon, instagibbs, Rasul, D.K., CongLiu, Britefury, Degrave, J.: Lasagne: First release, August 2015. http://dx.doi.org/10.5281/zenodo.27878
5. Dietterich, T.G., Bakiri, G.: Solving multiclass learning problems via error-correcting output codes. J. Artif. Intell. Res. **2**, 263–286 (1995)
6. Dong, L., Frank, E., Kramer, S.: Ensembles of balanced nested dichotomies for multi-class problems. In: Jorge, A.M., Torgo, L., Brazdil, P., Camacho, R., Gama, J. (eds.) PKDD 2005. LNCS(LNAI), vol. 3721, pp. 84–95. Springer, Heidelberg (2005). doi:10.1007/11564126_13
7. Duarte-Villaseñor, M.M., Carrasco-Ochoa, J.A., Martínez-Trinidad, J.F., Flores-Garrido, M.: Nested dichotomies based on clustering. In: Alvarez, L., Mejail, M., Gomez, L., Jacobo, J. (eds.) CIARP 2012. LNCS, pp. 162–169. Springer, Heidelberg (2012). doi:10.1007/978-3-642-33275-3_20
8. Frank, E., Kramer, S.: Ensembles of nested dichotomies for multi-class problems. In: Proceedings of the Twenty-First International Conference on Machine Learning, p. 39. ACM (2004)
9. Freund, Y., Schapire, R.E.: Game theory, on-line prediction and boosting. In: Proceedings of the Ninth Annual Conference on Computational Learning Theory, pp. 325–332. ACM (1996)
10. Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., Witten, I.H.: The weka data mining software: an update. ACM SIGKDD Explor. Newsl. **11**(1), 10–18 (2009)
11. Hastie, T., Tibshirani, R., et al.: Classification by pairwise coupling. Ann. Stat. **26**(2), 451–471 (1998)
12. Krizhevsky, A.: Learning multiple layers of features from tiny images. Master's thesis, University of Toronto, Toronto (2009)
13. Lichman, M.: UCI machine learning repository (2013). http://archive.ics.uci.edu/ml
14. Nadeau, C., Bengio, Y.: Inference for the generalization error. Mach. Learn. **52**(3), 239–281 (2003)
15. Rifkin, R., Klautau, A.: In defense of one-vs-all classification. J. Mach. Learn. Res. **5**, 101–141 (2004)
16. Rodríguez, J.J., García-Osorio, C., Maudes, J.: Forests of nested dichotomies. Pattern Recogn. Lett. **31**(2), 125–132 (2010)
17. Webb, G.I.: MultiBoosting: a technique for combining boosting and wagging. Mach. Learn. **40**(2), 159–196 (2000)