



The University of Sydney

A Benchmark for Classifier Learning

Technical Report Number 474

November, 1993

Zijian Zheng

ISBN 0 86758 900 0

**Basser Department of Computer Science
University of Sydney NSW 2006**

A BENCHMARK FOR CLASSIFIER LEARNING

ZIJIAN ZHENG

Basser Department of Computer Science, University of Sydney, NSW 2006
Email: zijian@cs.su.oz.au

ABSTRACT: Although many algorithms for learning from examples have been developed and many comparisons have been reported, there is no generally accepted benchmark for classifier learning. The existence of a standard benchmark would greatly assist such comparisons. Sixteen dimensions are proposed to describe classification tasks. Based on these, thirteen real-world and synthetic datasets are chosen by a set covering method from the UCI Repository of machine learning databases to form such a benchmark.

1 Introduction

Considerable advances have been made, in the field of classifier learning from examples, making it one of the most active research areas of machine learning. Many algorithms for this task have been developed and applied to problems from a variety of fields such as medical science [Detrano *et al.*, 1989], biology [Qian and Sejnowski, 1988], linguistics [Sejnowski and Rosenberg, 1987].

The task of zeroth-order classifier learning from examples is generally in the form:

Given: a set of examples, called the training set, in the form of a vector of attribute values and known class for each example.

Induce: a theory that can predict the classes of unseen cases of the same domain. The learned theory defines classes in terms of attribute values.

Algorithms for this task can be roughly divided into two types:

- symbolic algorithms, such as C4.5 [Quinlan, 1993], CART [Breiman *et al.*, 1984], PLS1 [Rendell, 1983, 1986], the AQ family of algorithms [Michalski and Chilausky, 1980]; and
- sub-symbolic algorithms, such as Boole [Wilson, 1987], back-propagation [Rumelhart *et al.*, 1986], Perceptron [Rosenblatt, 1962], IB1, IB2, IB3 [Aha *et al.*, 1991], MDLA, B, C [Cameron-Jones, 1992], Nearest Neighbor, Bayes, and Linear Discriminant [Duda and Hart, 1973].

All of these algorithms perform differently on different domains. In order to evaluate and compare the algorithms, many evaluation criteria have been developed. The three most commonly used ones are predictive accuracy on a test set, the size of learned theory, and learning time. A great amount of literature concerning evaluating or comparing classifier learning algorithms using these criteria (or some of them) has been published. One common problem is that only a few domains are used in each evaluation and different papers use different domains. It is very hard to judge an algorithm by seeing its performance on only a few arbitrarily selected domains.

This problem was met when developing and evaluating the constructive induction learning algorithm CI [Zheng, 1992]. At first, twelve arbitrarily selected domains from the UCI Repository of machine learning databases [Murphy and Aha, 1991] were used. It was found that constructing new attributes can improve the performance of decision trees significantly on four domains, can improve the performance but without high significance level on five domains, and gains almost nothing on the three other domains. With respect only to the former four domains, the algorithm appears to perform very well. Conversely, the algorithm appears to be very weak with respect to only the latter three.

Thus when evaluating or comparing classifier learning algorithms the following questions are raised: how many domains are reasonable and, furthermore, what kind of domains should be used? The objective of this paper is to address this problem by proposing a benchmark for classifier learning. Section 2 gives a set of dimensions for describing classification tasks. Then, Section 3 proposes a set of datasets to form a benchmark. Section 4 gives a simple demonstration of using the benchmark. Finally, Section 5 briefly describes related work.

2 A Taxonomy of Datasets

The aim of defining dimensions is to choose a set of domains that can represent as completely as possible domains for classifier learning, so all the aspects of domains that are related to classifier learning should be considered. Generally speaking, the dataset of a classification problem has three aspects: the form of the attributes, the form of the instances, and the form of the classes.

The following 16 dimensions will be used to describe datasets.

- Four concerning the attributes are the *type of attributes*, the *number of attributes*, the *number of different nominal attribute values*, and the *number of irrelevant attributes*.
- Five regarding the instances (attribute/class values) are the *dataset size*, the *dataset density*, the *level of noise in attribute values*, the *level of noise in class memberships or indeterminacy*, and the *frequency of missing attribute values*.
- Seven relating to the classes are the *number of classes*, the *default accuracy*, the

entropy, the *predictive accuracy*, the *relative accuracy*, the *average information score*, and the *relative information score*.

Some of them are straightforward and simple measures, such as the *dataset size*, the *number of classes*, and the *default accuracy*. Others are complex and sensitive measures, such as the *dataset density* and *entropy*. “*Sensitive*” here means that the dimension can reflect more detailed characteristics of a domain. For example, the simple dimension *number of classes* can only tell how many classes a domain has. However, the sensitive dimension *entropy* can reflect the class distribution of a domain. One reason for adopting the 16 dimensions is that different classifier learning algorithms behave differently along each dimension. For example, some systems can deal with, but some others have difficulty with: discrete values, continuous values, missing values, noise, more than two classes, or irrelevant attributes. Some systems slow down very rapidly as the number of training examples and/or the number of attributes increase, but others do not.

To make it easier to choose datasets for the benchmark, we define all the dimensions using nominal values. In this paper, we use terms, such as *small*, *low*, *medium*, *large*, and *high*, as values of some dimensions. *Small* and *low* mean *in the first quartile* of all the candidate values. *Large* and *high* mean *in the last quartile* of all the candidate values. *Medium* means *between the small (or low) and the large (or high)*.

Table 1. Datasets in the benchmark

Name	Description
Breast Cancer (W)	Medical diagnosis applied to breast cytology (Wisconsin)
Diabetes	Pima Indians diabetes database for diagnosing diabetes
Hepatitis	Predicting whether a patient will die from hepatitis
LED-24	LED display with 24 segments (17 irrelevant)
LED-7	LED display with 7 segments
Lymphography	Lymphography database
Monks-2	The second Monks’ problem
Mushroom	Mushrooms classified as poisonous or edible
NetTalk (Phoneme)	NetTalk Corpus for the phonetic transcription of the 1000 most common English words (prediction of phoneme)
Promoter	Promoter gene sequences (DNA)
Soybean	Large soybean database
Thyroid	Hypothyroid database (thyroid disease records)
Waveform-40	Waveform database with 40 attributes (19 irrelevant)

The dimensions and their nominal values are described below, with thresholds¹ and two examples taken from the final benchmark in Table 1 for each dimension value.

¹The thresholds are chosen based on an analysis of the domains in the UCI Machine Learning Databases. They correspond to the boundaries of the first and fourth quartile (as closely as possible).

1. *Type of attributes* (4 values):

- *Binary attributes only*, e.g. LED-24 and LED-7
- *Nominal attributes only*, e.g. Promoter and NetTalk (Phoneme)
- *Continuous attributes only*, e.g. Breast Cancer (W) and Diabetes
- *Mixed attributes*, e.g. Soybean (16 binary attributes, 19 nominal attributes) and Hepatitis (13 binary attributes, 6 continuous attributes)

2. *Number of attributes* (3 values):

- *Small* (less than 10), e.g. NetTalk (Phoneme) (7) and Breast Cancer (W) (9)
- *Medium* (between 10 and 30), e.g. Mushroom (22) and Hepatitis (19)
- *Large* (more than 30), e.g. Promoter (57) and Waveform-40 (40)

3. *Number of different nominal attribute values (#DNAV)* (3 values):

- *Small* (less than 5), e.g. Promoter (4) and Monks-2 (4)
- *Medium* (between 5 and 10), e.g. Soybean (7) and Lymphography (8)
- *Large* (more than 10), e.g. Mushroom (12) and NetTalk (Phoneme) (27)

4. *Number of irrelevant attributes (#IAtt)* (2 values):

For most real-world domains, it is quite difficult to know which attributes are irrelevant, but irrelevant attributes really exist in many real-world problems and affect the performance of classifier learning algorithms. Some algorithms are capable of identifying and deleting irrelevant attributes.

- *With irrelevant attributes*, e.g. LED-24 and Waveform-40
- *Without irrelevant attributes*, e.g. NetTalk (Phoneme) and LED-7

5. *Level of noise or indeterminacy*:

Noise is an important fact that affects the performance of learning algorithms. In real-world domains, such as medical domains, noise occurs due to errors introduced when measuring and diagnosing, and indeterminacy with respect to attributes. Noise in attribute-values and classes is inevitable, but the noise level often cannot be known. In the NetTalk domain [Dietterich *et al.*, 1990], because the window of 7 letters used is not large enough to uniquely determine the phoneme and stress of a letter, some instances with the same 7 letters might have different phonemes or stresses. We can therefore consider the class to be noisy or indeterminate. This dimension can be further divided into two:

- *In attribute values* (2 values):
 - *With noise*, e.g. LED-24 and Waveform-40

- *Without noise*, e.g. NetTalk (Phoneme) and Monks-2
- *In class memberships* (2 values):
 - *With noise or indeterminacy*, e.g. NetTalk (Phoneme) and Breast Cancer (W)
 - *Without noise or indeterminacy*, e.g. LED-24 and Waveform-40

6. *Frequency of missing attribute values* (3 values):

- *None*, e.g. Lymphography and NetTalk (Phoneme)
- *Few* (between 0 and 5.6%), e.g. Mushroom (1.39%) and Breast Cancer (W) (0.25%)
- *Many* (more than 5.6%), e.g. Soybean (9.78%) and Thyroid (6.74%)

7. *Dataset size* (3 values):

- *Small* (less than 210), e.g. Promoter (106) and Lymphography (148)
- *Medium* (between 210 and 3170), e.g. Diabetes (768) and Thyroid (3163)
- *Large* (more than 3170), e.g. NetTalk (Phoneme) (5438) and Mushroom (8124)

8. *Dataset density* (3 values):

Usually a classifier learning algorithm can learn a more accurate theory from a larger number of training examples than from fewer examples. However, because different domains have different sizes of description spaces, it is very difficult to say that a dataset containing more than N examples is large and containing less than N examples is small for some N . Therefore, besides the dataset size, we need the density of description space to characterize a dataset. This can be defined as:

$$\text{Density} = \frac{\text{Number-of-examples}}{\text{Size-of-description-space}}$$

$$\text{Size-of-description-space} = \prod_{i=1}^n N_i$$

where n is the number of attributes and N_i is, for the i -th attribute, the number of different values (if it is a binary or nominal attribute), or the number of different values in the dataset (if it is a continuous attribute).

- *Low* (less than 1.00×10^{-18}), e.g. Promoter (5.10×10^{-33}) and Waveform-40 (7.31×10^{-92})
- *Medium* (between 1.00×10^{-18} and 6.00×10^{-7}), e.g. Soybean (5.47×10^{-13}) and Lymphography (4.90×10^{-7})

- *High* (more than 6.00×10^{-7}), e.g. LED-24 (1.19×10^{-5}) and LED-7 (1.56)

9. *Number of classes* (3 values):

- *Binary*, e.g. Hepatitis and Promoter
- *Small or medium* (between 3 and 10), e.g. Lymphography (4) and Waveform-40 (3)
- *Large* (more than 10), e.g. Soybean (19) and NetTalk (Phoneme) (52)

10. *Default accuracy* (3 values):

The default accuracy (*the relative frequency of the most common class*), often used as a reference when discussing the predictive accuracy of a learning algorithm, reflects the hardness of a classification problem to some extent.

- *Low* (less than 40%), e.g. Soybean (13.7%) and NetTalk (Phoneme) (18.7%)
- *Medium* (between 40% and 75%), e.g. Lymphography (54.7%) and Breast Cancer (W) (65.5%)
- *High* (more than 75%), e.g. Hepatitis (79.4%) and Thyroid (95.2%)

11. *Entropy* (3 values):

For multi-class problems, besides considering the major class of a dataset like the *default accuracy*, the *entropy* takes the class distribution of a dataset into account². It can be defined as:

$$-\sum_{i=1}^N P(C_i) \times \log_2 P(C_i) \text{ bits}$$

where N is the number of classes and $P(C_i)$ is the prior probability of class C_i . It is the expected amount of information for classifying an example.

- *Low* (less than 0.80 bits), e.g. Hepatitis (0.73) and Thyroid (0.28)
- *Medium* (between 0.80 bits and 1.58 bits), e.g. Lymphography (1.23) and Diabetes (0.93)
- *High* (more than 1.58 bits), e.g. Soybean (3.84) and NetTalk (Phoneme) (4.72)

12. *Difficulty of the domain*:

The accuracy that the existing learning algorithms achieve can reflect the difficulty of a domain. If no algorithm can achieve high accuracy on a domain, this domain can be considered to be hard for learning. Here, we use the highest accuracy

²For 2 class problems, entropy can be calculated from default accuracy.

achieved by some existing algorithms, recorded in “Past Usage” in the UCI ML Databases, as *predictive accuracy*. If such an accuracy of a domain is not available, the accuracy of the well-known decision tree learning algorithm C4.5 on the domain is used. We define relative accuracy as

$$\text{Relative accuracy} = \frac{\text{Predictive accuracy} - \text{Default accuracy}}{100\% - \text{Default accuracy}} \times 100\%$$

where 100% in the denominator is the accuracy achieved by a perfect learning algorithm. We use the following two dimensions to discuss the *difficulty of the domain*.

- *Predictive accuracy* (3 values):
 - *Low* (less than 80.0%), e.g. Lymphography (78.4%) and LED-24 (70.0%)
 - *Medium* (between 80.0% and 98.5%), e.g. Hepatitis (83.0%) and Soybean (97.1%)
 - *High* (more than 98.5%), e.g. Mushroom (100.0%) and Monks-2 (100.0%)
- *Relative accuracy* (3 values):
 - *Low* (less than 52.0%), e.g. Hepatitis (17.5%) and Diabetes (39.3%)
 - *Medium* (between 52.0% and 88.5%), e.g. Lymphography (52.3%) and Breast Cancer (W) (84.9%)
 - *High* (more than 88.5%), e.g. Soybean (96.6%) and Mushroom (100.0%)

13. *Information score:*

The predictive accuracy is the most commonly used evaluation criterion, but it does not take into account the class prior probabilities and class distribution of a dataset. To overcome this shortcoming, Kononenko and Bratko [1991] introduce the average information score and relative information score criteria. They can be defined as:

$$I_{average} = \frac{1}{T} \times \sum_{j=1}^T I_{e_j}$$

$$I_{relative} = \frac{I_{average}}{Entropy_{test}} \times 100\%$$

where $I_{average}$ and $I_{relative}$ are the average and relative information score of a classifier respectively; $Entropy_{test}$ is the entropy of the test set; T is the number of test examples; I_{e_j} is the information score of the classifier on test example e_j , and defined as:

$$I_{e_j} = \begin{cases} -\log_2 P(C_{e_j}) + \log_2 P'(C_{e_j}) & \text{if } P'(C_{e_j}) \geq P(C_{e_j}) \\ -(-\log_2(1 - P(C_{e_j})) + \log_2(1 - P'(C_{e_j}))) & \text{if } P'(C_{e_j}) < P(C_{e_j}) \end{cases}$$

where C_{e_j} is the correct class of test example e_j ; $P(C_{e_j})$ is the prior probability of class C_{e_j} and $P'(C_{e_j})$ is the posterior probability returned by the classifier. The following two dimensions are used to describe the information score. Here the C4.5 is used as the classifier when calculating the information score.

- *Average information score* (3 values):
 - *Low* (less than 0.25 bits), e.g. Hepatitis (0.14) and Thyroid (0.24)
 - *Medium* (between 0.25 bits and 1.30 bits), e.g. Lymphography (0.61) and Mushroom (1.00)
 - *High* (more than 1.30 bits), e.g. Soybean (3.35) and LED-24 (1.64)
- *Relative information score* (3 values):
 - *Low* (less than 45.0%), e.g. Hepatitis (19.1%) and Promoter (42.1%)
 - *Medium* (between 45.0% and 85.5%), e.g. Lymphography (54.3%) and NetTalk (Phoneme) (79.4%)
 - *High* (more than 85.5%), e.g. Breast Cancer (W) (87.5%) and Soybean (91.4%)

When selecting dimensions, we require that all of them be learning algorithm independent. The first twelve dimensions discussed above completely satisfy this requirement. The “predictive accuracy” and “relative accuracy” dimensions depend on learning algorithms, but they depend on uniformly all kinds of classifier learning algorithms, rather than one special kind. Ideally, the “average information score” and “relative information score” dimensions should, like “predictive accuracy” and “relative accuracy”, depend on uniformly all kinds of classifier learning algorithms. Unfortunately we cannot access all these algorithms. We use the well-known algorithm C4.5 instead.

3 Data Sets

As classifier learning algorithms can be applied to many fields in science and technology, the number of possible domains is very large and the differences among them is great. It is impossible to evaluate or compare learning algorithms by using them on all possible domains. Here we use the dimensions discussed above to propose the least number of representative³ domains as a benchmark that covers the space described by the dimensions as completely and evenly as possible.

Although all the dimensions have been defined as having a few nominal values, the size of description space of domains is still 10^7 . Selecting one domain for each point in the description space is not practicable. A feasible way is to select at least one domain for each value of every dimension. In order to cover the dimensions adequately,

³*representative* here means that the domain has some common characteristics with several other domains, that is, it can be a prototype of a group of domains.

Table 2. Datasets in the benchmark

DataSet	Size	Missing Values	Noise Level		# Attributes				# IAtt	# DNAV	# Cl.
			Att.	Cl.	B	N	C	T			
Breast Cancer (W)	699	16(0.25)	yes	yes	0	0	9	9			2
Diabetes	768	0			0	0	8	8			2
Hepatitis	155	167(5.67)			13	0	6	19			2
LED-24	200	0	yes	no	24	0	0	24	17(70.8)		10
LED-7	200	0	yes	no	7	0	0	7	0		10
Lymphography	148	0			9	9	0	18		8	4
Monks-2	432	0	no	no	2	4	0	6	0	4	2
Mushroom	8124	2480(1.39)			4	18	0	22		12	2
NetTalk(Phoneme)	5438	0	no	yes	0	7	0	7	0	27	52
Promoter	106	0			0	57	0	57		4	2
Soybean	683	2337(9.78)	yes	yes	16	19	0	35		7	19
Thyroid	3163	5329(6.74)	yes	yes	18	0	7	25			2
Waveform-40	300	0	yes	no	0	0	40	40	19(47.5)		3

Table 2. Datasets in the benchmark - continued

DataSet	Density	Default Acc.	Entropy (bits)	Highest Acc.		Info. S. of C4.5	
				Pred.	Rel.	Average.	Rel.
Breast Cancer (W)	7.77×10^{-7}	65.5	0.93	94.8	84.9	0.81	87.5
Diabetes	1.12×10^{-13}	65.1	0.93	78.8	39.3	0.32	34.1
Hepatitis	1.28×10^{-12}	79.4	0.73	83.0	17.5	0.14	19.1
LED-24	1.19×10^{-5}	14.1	3.28	70.0	65.1	1.64	56.1
LED-7	1.56	14.1	3.28	71.0	66.2	1.92	66.2
Lymphography	4.90×10^{-7}	54.7	1.23	78.4	52.3	0.61	54.3
Monks-2	1.00	62.1	0.96	100.0	100.0	0.15	16.6
Mushroom	5.79×10^{-12}	51.8	1.00	100.0	100.0	1.00	100.0
NetTalk(Phoneme)	5.20×10^{-7}	18.7	4.72	84.1	80.4	3.70	79.4
Promoter	5.10×10^{-33}	51.5	1.00	76.3	51.1	0.41	42.1
Soybean	5.47×10^{-13}	13.7	3.84	97.1	96.6	3.35	91.4
Thyroid	1.32×10^{-17}	95.2	0.28	99.1	81.2	0.24	85.6
Waveform-40	7.31×10^{-92}	39.0	1.56	86.0	77.0	0.83	54.0

a requirement was imposed that every value of each dimension be represented by at least two datasets.

The aim of developing learning algorithms is to solving real-world problems, so datasets from real-world domains should be used when analyzing and/or comparing learning algorithms. Real-world datasets directly reflect the real situations of the real-world applications, but for most real-world domains, their exact concepts are unknown. Conversely, the true concepts of synthetic datasets are known and their characteristics such as the number of irrelevant attributes and noise level can be controlled. This makes it easier to analyze learning algorithms. Therefore, the benchmark should include a few synthetic datasets as well.

To minimize subjectivity when selecting domains, a two step process was followed:

Table 3. Accuracy of the IB1, C4.5, and CI2-2L algorithms on the benchmark

DataSet	IB1	C4.5	CI2-2L
Breast Cancer (W)	96.0	94.8	94.7
Diabetes	70.6	71.5	70.4
Hepatitis	81.9	78.2	82.1
LED-24	32.0	60.5	60.5
LED-7	71.0	69.5	70.5
Lymphography	82.4	78.4	81.1
Monks-2	70.4	<i>65.0</i>	72.7
Mushroom	100.0	100.0	100.0
NetTalk(Phoneme)	73.9	81.1	82.8
Promoter	83.0	76.3	81.0
Soybean	91.1	91.5	93.9
Thyroid	97.1	99.1	99.1
Waveform-40	67.7	69.4	72.7

- The candidates were all real-world domains and well-known synthetic domains from the UCI Repository of machine learning databases [Murphy and Aha, 1991]. To ensure that the benchmark does not contain relatively unknown datasets, only datasets that have been referenced in the last three Proceedings of the International Workshop/Conference on Machine Learning were considered. This dual constraint guarantees that all datasets are commonly known and easily accessible. Thirty-one such datasets were identified.
- A set covering algorithm was used to select the smallest subset of them that satisfies the coverage requirement. The algorithm was run 50 times with different random orderings of the 31 datasets and the frequency of occurrence of each dataset in the selected subsets was noted. With the datasets sorted by this frequency, the set covering algorithm was run once more to identify one subset. To make the covering more even, a simulated annealing algorithm was run on it, and the final subset was generated.

The result was a collection of thirteen datasets described briefly in Table 1. Table 2 details the description of the benchmark using concrete dimension values of datasets.

4 A Demonstration of Using the Benchmark

As a simple demonstration of using the benchmark, Table 3 gives the accuracy of the IB1 [Aha *et al.*, 1991], C4.5 [Quinlan, 1993], and CI2-2L [Zheng, 1992] algorithms obtained using a 10-fold cross-validation [Breiman *et al.*, 1984], on each domain of the benchmark⁴. IB1 is a simple instance-based learning algorithm. C4.5 is the latest version of ID3, a well-known decision tree learning algorithm. CI2-2L is a constructive induction learning

⁴On the Monks-2 domain, the training set and test set given by the problem designer were used, and only one trial was conducted.

algorithm that is based on C4.5. For each production rule generated from a decision tree having more than one condition, it uses as a new attribute the conjunction of two conditions that are nearest a leaf of the decision tree. The accuracy shown in boldface is significant higher than that in the preceding column at above the 95% level using an instance-based pairwised sign-test [Chatfield, 1978]. The accuracy shown in italics is significant lower than that in the preceding column at above the 95% level.

From Table 3, we can see that the performance of C4.5 is obviously much better than that of IB1 on the LED-24 domain. This is because 70% of attributes of LED-24 are irrelevant and C4.5 is more capable of tolerating irrelevant attributes than IB1. The Waveform-40 domain has irrelevant attributes (47%) too, but the performance of IB1 on it is much better in comparison. The reason might be that all attributes of Waveform-40 are continuous, treatment of continuous attributes in IB1 is better than that in C4.5, and continuous irrelevant attributes have less influence than nominal or binary attributes on IB1. Table 3 shows that CI2-2L achieves some accuracy improvement over C4.5 on 8 out of 13 domains. Three of them have a high significance level. By analyzing the characteristics of the domains in Table 2, we can see that CI2-2L performs well on the domains involving nominal attributes. One reason might be that a decision tree built using nominal attributes is likely to have more duplications of sequences of tests in different branches than a decision tree built using binary and/or continuous attributes and duplications are common near leaves of a tree, but this still needs further exploration.

5 Related Work

Almost all the developers of learning algorithms use some domains to demonstrate the performance of their algorithms and to compare theirs with other methods. For example, Mooney *et al.* [1989] use four domains to compare ID3 with perceptron and back-propagation methods and find that the predictive accuracy of the three algorithms is basically the same (except that back-propagation is more accurate on noisy data sets), but it is slower than IB3 and perceptron. Dietterich *et al.* [1990] compare ID3 and back-propagation on the NetTalk domain and discover that the back-propagation with its numerical parameters can capture statistical information that is not captured by ID3. Weiss and Kapouleas [1989] use four domains to compare statistical, connectionist, and machine learning classification algorithms. Schaffer [1992] uses five domains to compare four classification methods and shows that cross-validation might be an approach to applying partial information about appropriate methods for classification. Holte [1993] uses fifteen domains to compare 1R with C4.5 and concludes that on most data sets the best of very simple rules based on a single attribute is as accurate as the rules created by the majority of machine learning systems. Thrun *et al.* [1991] use the MONK's problems to evaluate 19 symbolic and non-symbolic learning algorithms. To the best of our knowledge, however, there is no generally accepted benchmark for classifier learning.

Brazdil *et al.* [1993] use twenty-three domains from the Esprit Project StatLog to

compare and characterize twenty-two different learning algorithms by using meta level learning. The test results of the algorithms on the domains together with domain descriptions are used as input data for the meta level learning and, for each algorithm, rules predicting the applicability of the algorithm are synthesized. The domains are characterized by using three simple measures such as the number of examples, six statistical measures such as SD-ratio (geometric mean ratio of standard deviations of the individual populations to the pooled standard deviation), and eleven information based measures such as Cx_attrib (complexity of attributes based on the notion of information), Cx_class (complexity of classification scheme), and Gain_attrib (the total gain provided by the attributes). Among these measures, three simple and eleven information based measures reflect characters on attributes, instances, and classes of data sets. The three simple and six information based measures are included in our dimension set. The other four information based measures are not directly included in our dimension set, but what they measure can be indirectly reflected by our dimensions. For example, besides the number of attributes, Cx_attrib takes the number of different nominal attribute values into account. Correspondingly, we have the *number of different nominal attribute values* dimension. Although Gain_attrib is not included in our dimension set, what Gain_attrib measures can be indirectly reflected by *information score* because C4.5 is used to compute *information score* on a data set. C4.5 uses gain or gain ratio as test selection criterion to build trees. Goodness of a tree built on a data set can reflect the gain provided by the attributes of the data set to some extent. The only information based measure not included in our dimension set is CostU (cost unbalance), because it depends on information not contained in the data, i.e. a misclassification cost matrix. The statistical measures characterize data sets from the statistical point of view. They are biased toward distinguishing performances of statistical classifier learning algorithms by checking whether statistical assumptions such as normality that form the basis of parametric methods are violated by the data, while we expect that all dimensions are algorithm independent. Therefore, our dimension set does not include their statistical measures.

6 Conclusion

A benchmark for classifier learning is very important for evaluating or comparing algorithms for learning from examples. In this paper, sixteen dimensions are developed to describe classification tasks. Based on these, thirteen real-world and synthetic datasets are chosen from the UCI Repository of machine learning databases to form a benchmark.

The benchmark provides a basis for evaluating and comparing algorithms for learning from examples. It also makes it possible to characterize any given learning algorithm. With the benchmark, we can pinpoint particular strengths and/or weaknesses of different classifier learning algorithms with respect to measures such as accuracy, theory size, and learning time.

7 Acknowledgements

This research was supported by an ARC grant (to Ross Quinlan) and by a research agreement with Digital Equipment Corporation. The author is supported by EMSS scholarship. Many thanks to Ross Quinlan for his advice and suggestions, particularly about doing this work. I'd also like to thank Mike Cameron-Jones, Alen Varšek, and Kai Ming Ting for their helpful discussions and suggestions. Finally, P.M. Murphy and D. Aha are gratefully acknowledged for creating and managing the UCI Repository of machine learning databases.

References

- [Aha *et al.*, 1991] D.W. Aha, D. Kibler, and M.K. Albert, Instance-based learning algorithms, *Machine Learning*, **6**, 37-66, 1991.
- [Brazdil *et al.*, 1993] P. Brazdil, J. Gama, and B. Henery, Comparison of ML and statistical approaches using meta level learning, *Workshop Notes on Real-World Applications of Machine Learning, European Conference on Machine Learning*, 1993.
- [Breiman *et al.*, 1984] L. Breiman, J.H. Friedman, R.A. Olshen, and C.J. Stone, *Classification And Regression Trees*, Belmont, CA: Wadsworth, 1984.
- [Chatfield, 1978] C. Chatfield, *Statistics for Technology: A Course in Applied Statistics*, Chapman and Hall, 1978.
- [Cameron-Jones, 1992] R.M. Cameron-Jones, Minimum description length instance-based learning, *Proceedings of Australian Joint Conference on Artificial Intelligence*, World Scientific Publisher, 368-373, 1992.
- [Detrano *et al.*, 1989] R. Detrano, A. Janosi, W. Steinbrunn, M. Pfisterer, J. Schmid, S. Sandhu, K. Guppy, S. Lee, and V. Froelicher, International application of a new probability algorithm for the diagnosis of coronary artery disease, *American Journal of Cardiology*, **64**, 304-31, 1989.
- [Dietterich *et al.*, 1990] T.G. Dietterich, H. Hild, and G. Bakiri, A comparative study of ID3 and backpropagation for English text-to-speech mapping, *Proceedings of the Seventh International Workshop on Machine Learning*, 24-31, 1990.
- [Duda and Hart, 1973] R. Duda and P. Hart, *Pattern Classification and Scene Analysis*, New York: Wiley, 1973.
- [Holte, 1993] R.C. Holte, Very simple classification rules perform well on most datasets, *Machine Learning*, **11**, 63-90, 1993.

- [Kononenko and Bratko, 1991] I. Kononenko and I. Bratko, Information-based evaluation criterion for classifier's performance, *Machine Learning*, **6**, 67-80, 1991.
- [Lowe, 1993] D.G. Lowe, Similarity metric learning for a variable-kernel classifier, *Technical Report, Computer Science Department, University of British Columbia, Canada*, 1993.
- [Michalski and Chilausky, 1980] R.S. Michalski and R.L. Chilausky, Learning by being told and learning from examples, *International Journal of Policy Analysis and Information Systems*, **4**, 125-160, 1980.
- [Mooney *et al.*, 1989] R. Mooney, J. Shavlik, G. Towell, and A. Gove, An experimental comparison of symbolic and connectionist learning algorithms, *Proceedings of the Eleventh International Joint Conference on Artificial Intelligence*, 775-780, Morgan Kaufman, 1989.
- [Murphy and Aha, 1991] P.M. Murphy and D.W. Aha, *UCI Repository of machine learning databases* [Machine-readable data repository]. Irvine, CA: University of California, Department of Information and Computer Science, 1991.
- [Qian and Sejnowski, 1988] N. Qian and T.J. Sejnowski, Predicting the secondary structure of globular proteins using neural network models, *Journal of Molecular Biology*, **202**, 865-884, Academic Press, 1988.
- [Quinlan, 1993] J.R. Quinlan, *C4.5: Programs for Machine Learning*, San Mateo, CA: Morgan Kaufman, 1993.
- [Rendell, 1983] L.A. Rendell, A new basis for state-space learning systems and a successful implementation, *Artificial Intelligence*, **20**, 369-392, 1983.
- [Rendell, 1986] L.A. Rendell, Induction, of and by probability, *Uncertainty in Artificial Intelligence*, Amsterdam: Elsevier Science Publishers, 1986.
- [Rosenblatt, 1962] F. Rosenblatt, *Principles of Neuradynamics*, Spartan, New York, 1962.
- [Rumelhart *et al.*, 1986] D.E. Rumelhart, G.E. Hinton, and J.R. Williams, Learning internal representations by error propagation, *Parallel Distributed Processing*, **Vol. 1**, MIT Press, Cambridge, MA, 318-362, 1986.
- [Schaffer, 1992] C. Schaffer, Selecting a classification method by cross-validation, *Technical Report, Department of Computer Science, CUNY-Hunter College*, 1992.

- [Sejnowski and Rosenberg, 1987] T.J. Sejnowski and C.R. Rosenberg, Parallel networks that learn to pronounce English text, *Complex Systems*, **1**, 145-168, 1987.
- [Thrun *et al.*, 1991] S.B. Thrun, J. Bala, E. Bloedorn, I. Bratko, B. Cestnik, J. Cheng, K. De Jong, S. Dżerpski, S.E. Fahlman, D. Fisher, R. Hamann, K. Kaufman, S. Keller, I. Kononenko, J. Kreuziger, R.S. Michalski, T. Mitchell, P. Pachowicz, Y. Reich, H. Vafaie, W. Van de Welde, W. Wenzel, J. Wnek, and J. Zhang, The MONK's problems - a performance comparison of different learning algorithms, *Technical Report: CMU-CD-91-197*, Carnegie Mellon University, 1991.
- [Towell *et al.*, 1991] G.G. Towell, M.W. Craven, and J.W. Shavlik, Constructive induction in knowledge-based neural networks, *Proceedings of the Eighth International Conference on Machine Learning*, 213-217, Morgan Kaufmann, 1991.
- [Weiss and Kapouleas, 1989] S.M. Weiss and I. Kapouleas, An empirical comparison of pattern recognition, neural nets, and machine learning classification methods, *Proceedings of the Eleventh International Joint Conference on Artificial Intelligence*, 781-787, Morgan Kaufman, 1989.
- [Zheng, 1992] Z. Zheng, Constructing conjunctive tests for decision trees, *Proceedings of Australian Joint Conference on Artificial Intelligence*, World Scientific Publisher, 355-360, 1992.