Is an Ordinal Class Structure Useful in Classifier Learning?

Jens C. Hühn and Eyke Hüllermeier[*]
Department of Mathematics and Computer Science
Marburg University, Germany
Hans-Meerwein-Str., 35032 Marburg, Germany
{huehnj,eyke}@informatik.uni-marburg.de

Tracking Number IJDMMM-8118

Abstract

In recent years, a number of machine learning algorithms have been developed for the problem of ordinal classification. These algorithms try to exploit, in one way or the other, the order information of the problem, essentially relying on the assumption that the ordinal structure of the set of class labels is also reflected in the topology of the instance space. The purpose of this paper is to investigate, on an experimental basis, the validity of this assumption. Moreover, we seek to answer the question to what extent existing techniques and learning algorithms for ordinal classification are able to exploit order information, and which properties of these techniques are important in this regard.

Keywords: ordinal classification, binary decomposition, nested dichotomies, pairwise classification.

1 Introduction

The problem of *ordinal classification*, also called *ordinal regression* in statistics, has received increasing attention in the machine learning field in recent years (Frank and Hall, 2001; Chu and Keerthi, 2005; Cardoso et al., 2005; Yu et al., 2006; Cardoso and da Costa, 2007; Babaria et al., 2007). In ordinal classification, the set of class labels

[*] Corresponding author (phone: ++49 6421 2821569, fax: ++49 6421 2821573)

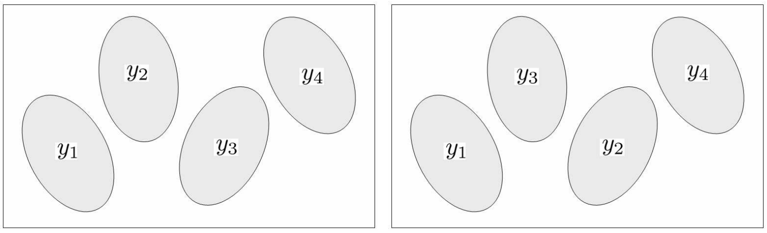

Figure 1: On the left, the distribution of classes in the instance space is in well agreement with the class order $y_1 \prec y_2 \prec y_3 \prec y_4$, while this is not the case in the right situation.

$\mathcal{Y} = \{y_1 \ldots y_m\}$ is endowed with a natural (total) order relation: $y_1 \prec y_2 \prec \cdots \prec y_m$. This distinguishes ordinal from conventional classification, where $\mathcal{Y}$ is unordered. As examples, consider learning to predict the category of a hotel (from 1 to 5 stars), the priority level of emails, or the customer satisfaction on a discrete scale ranging from, e.g., `poor` to `excellent`.

From a learning point of view, the ordinal structure of $\mathcal{Y}$ is *additional* information that a learner should of course try to exploit, and this is what existing methods for ordinal classification essentially seek to do (Frank and Hall, 2001; Cardoso and da Costa, 2007). The basic assumption in this regard is that the ordinal structure of $\mathcal{Y}$ is also present in the instance space $\mathcal{X}$, where it is reflected by the topology of the class distributions. Or, stated differently, the ordinal class structure induces an *ordinal instance structure*. Fig. 1 illustrates this idea: In the first scenario (left picture), the topology of $\mathcal{X}$ is in well agreement with the ordinal structure of $\mathcal{Y}$, which is not the case in the second situation (right picture).

The above assumption is most explicitly expressed by ordinal classification methods which are based on binary decomposition techniques, that is, techniques for transforming a polychotomous problem involving m classes into a set of binary problems: Given an ordinal instance structure, it is more reasonable and presumably simpler to discriminate, for example, the "low" classes $\{y_1 \ldots y_k\}$ from the "high" classes $\{y_{k+1} \ldots y_m\}$ than to discriminate an arbitrary subset of $\mathcal{Y}$ from its complement. More generally, it appears reasonable to restrict to "ordered" binary decompositions, where a binary problem involving meta-classes $\mathcal{Y}_1, \mathcal{Y}_2 \subset \mathcal{Y}$ is ordered if $y_i \prec y_j$ for all $(y_i, y_j) \in \mathcal{Y}_1 \times \mathcal{Y}_2$ or $y_i \succ y_j$ for all $(y_j, y_i) \in \mathcal{Y}_1 \times \mathcal{Y}_2$. One may even argue that this property may provide the basis for a definition of the value of ordinal structure: Roughly, the value of order information equals the (expected) increase in performance when solving the problem for ordered instead of unordered decompositions.

Of course, it is not at all self-evident that the assumption of an ordinal instance structure will hold in practice and, therefore, that ordinal classification techniques are actually effective. By effectiveness, we mean that these techniques are able to exploit the ordinal information, if any, contained in a problem. The purpose of this paper is to investigate empirically whether or not ordinal classifiers are effective in this sense. Our analysis is based on the following key idea: If an ordinal classifier is effective in the above sense, then its expected performance on the true ordinal problem should be better than its expected performance on a distorted problem in which the label set is given by an arbitrary permutation (renaming) of $\mathcal{Y}$.

In this regard, it is important to note that the effectiveness or, say, the degree to which a classifier benefits from ordinal classification, is likely to depend on the flexibility of the classifier. For example, for a linear classifier it is easy to separate classes $\{y_1, y_2\}$ from $\{y_3, y_4\}$ in the first scenario in Fig. 1 but impossible in the second one. Therefore, a linear classifier will strongly benefit from the ordinal instance structure. The benefit of a decision tree, on the other hand, will be much smaller: Indeed, the first problem is also simpler for this learner, but the second one is still feasible.

The above considerations give rise to the following conjectures that we shall try to answer by means of suitable experiments:

- Knowledge about the ordinal structure of the label set $\mathcal{Y}$ is useful in a classification setting, and ordinal classifiers can effectively exploit this knowledge.
- The degree to which a learner benefits from an ordinal structure depends on its flexibility: Complex methods producing models with flexible decision boundaries will benefit less than methods producing simple decision boundaries.

As mentioned previously, ordinal classifiers based on binary decomposition techniques appear to be especially suitable for analyzing these hypotheses. Therefore, we shall focus on these techniques, to be surveyed in Section 2. Our experimental setting will then be outlined in Section 3, and the results will be presented in Section 4. The paper ends with a summary and discussion in Section 5.

2 Algorithms for Ordinal Classification

This section gives a brief introduction to the learning algorithms that we used in the experiments. The main purpose is to convey the basic ideas underlying the approaches. For more detailed information, we give pointers to the literature.

2.1 A Simple Approach To Ordinal Classification

A simple and intuitively appealing approach to ordinal classification has been proposed by Frank and Hall (2001). The idea is to decompose the original problem involving m classes $\mathcal{Y} = \{y_1 \ldots y_m\}$ into $m-1$ binary problems. The i-th problem is defined by the meta-classes $\{y_1 \dots y_i\}$ and $\{y_{i+1} \dots y_m\}$ playing the role, respectively, of the negative and positive class in a binary problem.

Let $\mathcal{M}_i$, $i = 1 \dots m-1$, denote the model learned on the training data for the i-th problem (i.e., considering examples with labels in $\{y_1 \dots y_i\}$ as negative and the others as positive examples). Given a query instance $\boldsymbol{x}$, a prediction $\mathcal{M}_i(\boldsymbol{x})$ is interpreted as an estimation of the probability that the class of $\boldsymbol{x}$, denoted $y(\boldsymbol{x})$, is in $\{y_{i+1} \dots y_m\}$, that is, an estimation of the probability $\Pr(y(\boldsymbol{x}) \succ y_i)$. Consequently, the models must guarantee outputs in the unit interval.

From the above probabilities, a probability distribution on $\mathcal{Y}$ is then derived as follows:

$$\begin{aligned}
\Pr(y(\boldsymbol{x}) = y_1) &= 1 - \Pr(y(\boldsymbol{x}) \succ y_1) \\
\Pr(y(\boldsymbol{x}) = y_i) &= \max\{\, \Pr(y(\boldsymbol{x}) \succ y_{i-1}) - \Pr(y(\boldsymbol{x}) \succ y_i),\, 0\,\} \\
\Pr(y(\boldsymbol{x}) = y_m) &= \Pr(y(\boldsymbol{x}) \succ y_{m-1})
\end{aligned}$$

Eventually, the class with the highest probability is predicted. As mentioned in the introduction, this approach strongly exploits the idea that “reasonable” decompositions of the class labels are produced by ordinal splits partitioning $\mathcal{Y}$ into a lower and an upper part.

2.2 Ensembles of Nested Dichotomies

A *nested dichotomy* is a binary tree that partitions the label set $\mathcal{Y}$ in a recursive way. The root of the tree is associated with the whole set $\mathcal{Y}$, while the leaf nodes correspond to single classes. Moreover, each inner node is associated with a binary classification problem, namely to discriminate between the two respective meta-classes of the child nodes. The output of a corresponding model, for a query input $\boldsymbol{x}$, is interpreted as a conditional probability of the form

$$p = \Pr(y(\boldsymbol{x}) \in \mathcal{Y}_2 \,|\, y(\boldsymbol{x}) \in \mathcal{Y}_1 \cup \mathcal{Y}_2),$$

where $\mathcal{Y}_1$ and $\mathcal{Y}_2$ denote, respectively, the meta-classes of the two child nodes (and hence $\mathcal{Y}_1 \cup \mathcal{Y}_2$ the meta-class of the inner node itself). Consequently, the probabilities of the individual classes y_i can be derived quite elegantly, namely by multiplying the probabilities along the path from the root of the tree to the leaf node for y_i. Nested dichotomies have been investigated for a long time in statistics.

Obviously, there are many ways to partition $\mathcal{Y}$ in a recursive way, and indeed, the prediction accuracy of a model may strongly depend on the choice of the concrete dichotomy. Frank and Kramer (2004) have therefore combined nested dichotomies with ensemble techniques. An *ensembles of nested dichotomies* (END) consists of a set of randomly generated nested dichotomies, the predictions of which are combined

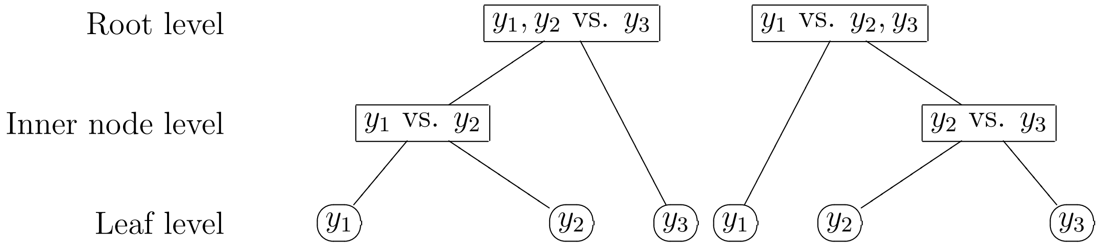

Figure 2: Two ordinal dichotomy trees for a 3-class problem.

by averaging the respective probability distributions. For this approach, the authors have reported excellent classification accuracy.

ENDs can be applied to conventional classification problems, and indeed, no restrictions are made for the splitting of label sets into subsets. In the case of ordinal classification, however, it again seems reasonable to restrict to ordered splits; see Fig. 2 for an example of an ordinal nested dichotomy.

Even though the number of different dichotomies is significantly smaller for the ordinal than for the general case, it may still become huge for a large number of classes m. More concretely, it can be shown by simple combinatorial arguments that the number is $(3^m - (2^{m+1} - 1))/2$ for the general case, which is reduced to $(m^3 - m)/6$ for the ordinal case. The computation of all dichotomies may thus become infeasible for large m. Frank and Kramer (2004) found that averaging over 20 randomly generated dichotomies is "sufficient to get close to optimum performance". In our experiments, we shall stick to this rule of thumb.

2.3 Pairwise Classification

Another popular binarization technique is the all-pairs approach, also called round robin learning (Fürnkranz, 2002a,b), which trains a separate model $\mathcal{M}_{i,j}$ for each pair of classes $(y_i, y_j) \in \mathcal{Y} \times \mathcal{Y}$, $1 \leq i < j \leq m$; thus, a total number of $m(m-1)/2$ models is needed. $\mathcal{M}_{i,j}$ is intended to discriminate between classes y_i and y_j. At classification time, a query $\boldsymbol{x}$ is submitted to all models, and each prediction $\mathcal{M}_{i,j}(\boldsymbol{x})$ is interpreted as a vote for a label. More specifically, assuming $s_{i,j} = \mathcal{M}_{i,j}(\boldsymbol{x}) \in [0, 1]$, the *weighted voting* techniques interprets $s_{i,j}$ and $1 - s_{i,j}$ as weighted votes for classes y_i and y_j, respectively, and predicts the class with the highest sum of weighted votes.

For the following reason, pairwise classification is an interesting baseline in our context: It produces binary problems that are (trivially) ordered and hence "reasonable" from an ordinal classification point of view, and yet it does not exploit any ordinal information. Fürnkranz (2002b) found that pairwise classification, using decision trees as base learners, is indeed competitive to the approach of Frank and Hall in terms of classification accuracy. As will be seen later on, our results are in agreement with this finding.

3 Experimental Setup

3.1 Classifiers

As learning algorithms, we used the three meta-techniques outlined in the previous section, namely the approach by Frank and Hall (FH), ensembles of ordered nested dichotomies (EOND), and round robin (RR). These techniques were instantiated, respectively, with the following base learners: Decision trees (C4.5) (Quinlan, 1993), logistic regression (LR), voted Perceptrons (VP) (Freund and Schapire, 1999), and support vector machines (SVM) with linear and RBF kernels (Vapnik, 1999). Except for ensembles of nested dichotomies, that we implemented by ourselves, all these algorithms are available in the WEKA machine learning framework (Witten and Frank, 2005) and were used in their default settings.[1]

3.2 Data Sets

Due to a lack of ordinal benchmark data, several previous studies, including (Frank and Hall, 2001; Fürnkranz, 2002b), have resorted to discretized regression data for experimental purposes. This is reasonable and has the advantage that, by changing the discretization, ordinal data can be produced in a quite flexible manner, for example with different numbers of classes. On the other hand, since regression data is not only ordinal but even numerical, one may expect that its ordinal structure is even stronger developed than for truly ordinal classification data. As this is critical for the purpose of our study, we complemented this type of data with truly ordinal classification data.

For the first part of the experiments we used the same 29 UCI regression data sets that were used in (Frank and Hall, 2001; Fürnkranz, 2002b); see Table 3. To obtain an ordinal class attribute, the numerical output attributes were discretized into m classes using equi-frequency binning (thereby producing classes of equal size). We only report results for $m = 5$ and $m = 7$, since other values for m (between 3 and 10) produced quite similar results. In addition, we collected 15 truly ordinal data sets[2]; see Table 4. These data sets were collected from several repositories (Asuncion and Newman, 2007; Meyer and Vlachos, 2007; David, 2008; Harvey, 2007; Bulloch, 2007; Barker, 2007; Townsend, 2007).

[1] For SVMs, the classification scores are transformed into (pseudo-)probabilities using a logistic regression technique (Platt, 1999).

[2] The dataset Nursery 1k is a random sample of size 1000 from the Nursery data.

3.3 Measuring the Value of Ordinal Information

As mentioned previously, the idea of our experimental study is to compare the performance of a learner on the original classification problem with its performance when being applied to a random permutation of the label set $\mathcal{Y}$. By a random permutation, we mean that the classes are re-numbered (from 1 to m) in a random way. Assuming that the true class order does indeed contain important information, and that the learner is able to exploit this information, the performance on the original problem should be better than on the distorted ones, at least on average.

More specifically, we compare the performance, in terms of classification accuracy, on the original problem with the performance on 10 random permutations (except for the cases $m = 3$ and $m = 4$, where we consider all 2 and 11 possible permutations, respectively).[3] As a measure of effectiveness or, say, the *value of order information* (VOI), we derive the rank r of the true class order in this set of problems, normalized to the range $[0, 1]$. More precisely, r is the percentage of random class orders for which the learner produces results that are worse than for the true order. In particular, $r = 1$ means that the best result was obtained for this order, while $r = 0$ means that the true class order has yielded the worst performance.

Finally, we test the statistical (null) hypothesis that $r \leq 0.5$ against the (alternative) hypothesis $r > 0.5$, using a win/loss sign test according to Demšar (2006).

4 Experimental Results

The experiments were conducted by randomly splitting each dataset into 2/3 for training and 1/3 for testing, and deriving the classification accuracy for each learner. To obtain more accurate estimates of expected performance, this procedure was repeated 50 times and the classification accuracy was averaged. Due to the high computational complexity, we omitted SVM as base learners in the experiments on regression data.

4.1 The Value of Order Information

For the regression data, the VOI as defined above is given in Tables 5–6. As can be seen, the values are significantly higher than 0.5 throughout, which means that the problems do indeed contain important order information, and that all learners are able to exploit this information. Regarding the base learners, it can furthermore be seen that LR gains more than C4.5 and VP. This provides strong evidence for our second conjecture, since LR fits a linear model and, therefore, is less flexible than

[3] We ignore the reversal of a class order, as it gives the same result.

C4.5 and VP. Finally, it is worth mentioning that FH seems to benefit stronger than EOND, suggesting that the ordinal structure is even more important for the former than for the latter technique. A likely explanation for this finding is that EOND is an ensemble technique and, hence, additionally benefits from an averaging effect which makes it less sensitive.

The VOI for the ordinal data sets is shown in Table 7. Even though the values again exceed 0.5 most of the time, a significant gain is only observed for the linear base learners. Again, this confirms our second conjecture. Moreover, since the VOI values are obviously smaller than for the regression data, the results confirm our presumption that discretized regression data exhibits an even stronger developed ordinal structure than truly ordinal data.

4.2 Absolute Classification Accuracy

In the previous study, we looked at each learner separately and investigated to what extent it benefits from order information. In a second study, we compared, in terms of classification accuracy, the ordinal classification techniques (meta-classifiers) with each other, though again separately for each base learner. Following Demšar (2006), we first applied a Friedman test (Friedman, 1937, 1940) on the hypothesis that all methods perform equally well. In case this hypothesis is rejected, a Nemenyi test (Nemenyi, 1963) was applied as post-hoc test to find significant differences between pairs of methods.

The classification accuracies for the regression data are given in Tables 8–9, with the significance of differences indicated at the bottom. A first result is that, on average, EOND seems to be the strongest meta-classifier. This result is hardly surprising in light of the fact that, in comparison to the other techniques, nested dichotomies use significantly more binary models (see Table 1). Again, however, there seems to be an influence of the base learner: While EOND is especially strong for decision trees, it is only en par with RR for logistic regression and voted perceptrons, and sometimes RR is even better. The results for the ordinal data sets, given in Table 10, show the same tendency. However, just like in the first study, they are less distinct. In fact, the three meta classifiers are more or less en par, and the differences, if any, are statistically not significant.

To make the comparison more fair, we considered two other versions of EOND that use less than 20 dichotomies: EOND-1 uses only a single (random) dichotomy and, therefore, exactly the same number of binary models as FH. By aggregating over 4 dichotomies, EOND-4 uses approximately the same number of binary models as RR; actually, it still uses slightly more on average. Note that the number of models is only a rough indicator of the amount of training effort invested by the learners. Alternatively, for example, one could count the total number of training examples. This would be advantageous for RR, as it only trains on small problems

m	EOND	FH	RR
3	4	2	3
5	20	4	10
7	41	6	21
9	64	8	36

Table 1: The number of binary models used by the meta classifiers. Since for EOND, this number depends on the random selection of the (maximally) 20 dichotomies, we report empirical averages. Binary models occurring in more than one dichotomy are only counted once.

	C4.5	LR	VP
EOND-1 vs FH	13:16	16:13	20:9
EOND-4 vs RR	23:6	11:18	10:19

Table 2: Win-loss statistics for variants of EOND.

(involving two classes) and disadvantageous for FH which always trains on the full set of examples; EOND is in-between, as it trains on problems of different size.

The classification accuracies for the regression data ($m = 5$) are shown in Fig. 11 and the win-loss statistics are summarized in Table 2. As can be seen, EOND is still superior for decision trees as base learners, but the situation reverses for LR and VP. This observation might be explained by the flexibility of the base learners in conjunction with the difficulty of the binary problems: RR only solves "small" classification problems involving two classes $y_i, y_j \in \mathcal{Y}$, whereas EOND has to solve binary problems that involve more classes $y_i \in \mathcal{Y}$, and the problems of FH even involve all classes. Now, since bigger meta-classes will usually call for more complex models (decision boundaries), flexible classifiers such as decision trees are advantageous for EOND, and even more so for FH. RR, on the other hand, has to solve only simple problems and, therefore, has a comparative advantage for less flexible base learners. Thus, RR has an "advantage of simplicity" which, however, disappears for learners that are flexible enough. In that case, EOND has an advantage, as it uses more training information (in terms of the number of examples). Regarding the comparison between EOND and FH, the situation is just reversed: FH uses even more training information but has to solve more difficult problems, so it profits from flexible models to an even greater extent.

5 Discussion and Conclusions

In summary, the results of our experimental studies confirm the conjectures that were raised at the beginning of the paper. More specifically, we can draw the following conclusions:

- Learning techniques for ordinal classification are indeed able to exploit order information about classes.
- Yet, the value of this information is usually not as high for “truly” ordinal data sets as it is for discretized regression data, which is often used in experimental studies.
- Another important factor is the flexibility of the learner. Roughly speaking, the less flexible a learner, the more it benefits from an ordinal structure.

As we have furthermore seen, the flexibility of the base learners is also important for the effectiveness of the meta-techniques investigated in this paper. Roughly speaking, these techniques produce binary problems of different complexity and, therefore, profit from a flexible base learner to a greater or lesser extent. An important implication is that meta-techniques cannot be compared independently of the base learner.

Regarding the meta-techniques, the role of the all-pairs decomposition scheme (round robin learning) seems to be especially interesting. This approach turned out to be fully competitive, even though it does not explicitly exploit ordinal structure (and even uses the smallest amount of training information in terms of the total number of training examples). However, by training only on pairs of classes, it is trivially consistent with each ordinal structure. On the one hand, one may say that it exploits ordinal structure in an implicit way whenever this is possible, but as its binary problems are not explicitly tailored toward the assumption of an ordinal structure, it does also not deteriorate when this assumption is invalid. On the other hand, one may also argue that it is unable to exploit ordinal structure, simply because such a structure does not become visible when looking only at pairs of classes. Anyway, it is competitive to more sophisticated decomposition schemes, especially for simple base learners, and therefore seems to be a good compromise.

Dataset	# inst.	# num.	# nom.
2D Planes	40768	10	0
Abalone	4177	7	1
Ailerons	13750	40	0
Auto MPG	398	4	3
Auto Price	159	14	1
Bank 32NH	8192	32	0
Bank 8FM	8192	8	0
Boston Housing	506	12	1
California Housing	20640	8	0
CPU Act	8192	21	0
CPU Small	8192	12	0
Delta Ailerons	7129	5	0
Delta Elevators	9517	6	0
Diabetes	43	2	0
Elevators	16599	18	0
Friedman Artificial	40768	10	0
House 16H	22784	16	0
House 8L	22784	8	0
Kinematics	8192	8	0
Machine CPU	209	6	0
MV Artificial	40768	7	3
Pole Telecom	15000	48	0
Pumadyn 32H	8192	32	0
Pumadyn 8NH	8192	8	0
Pyrimidines	74	27	0
Servo	167	0	4
Stocks	950	9	0
Triazines	186	60	0
Wisconsin Breast C.	194	32	0

Table 3: The 29 regression data sets used for the experiments. The second column contains the number of instances, the third column the number of numeric and the last column the number of nominal attributes (excluding the target to be predicted).

Dataset	# inst.	# classes.	# num.	# nom.
Asbestos	83	3	1	2
Balance Scale	625	3	4	0
CMC	1473	3	2	7
Pasture Production	36	3	21	1
Postoperative	90	3	0	8
Squash (unstored)	52	3	20	3
Car	1728	4	0	6
Grub Damage	155	4	2	6
Nursery 1k	1000	4	0	9
SWD	1000	4	100	0
Bondrate	57	5	4	7
Eucalyptus	736	5	14	5
LEV	1000	5	40	0
ERA	1000	9	40	0
ESL	488	9	40	0

Table 4: The 15 data sets with a truly ordinal class attribute. The second column contains the number of instances, the third column the number of numeric and the last column the number of nominal attributes (excluding the target to be predicted).

$m = 5$	EOND			FH		
Dataset	C4.5	LR	VP	C4.5	LR	VP
2D Planes	0.90	1.00	1.00	1.00	1.00	1.00
Abalone	0.30	1.00	0.60	1.00	1.00	0.90
Ailerons	1.00	1.00	0.10	1.00	1.00	0.70
Auto MPG	0.90	1.00	1.00	1.00	1.00	0.90
Auto Price	1.00	1.00	0.50	1.00	1.00	0.70
Bank 32NH	1.00	1.00	1.00	1.00	1.00	1.00
Bank 8FM	0.40	1.00	1.00	1.00	1.00	1.00
Boston Housing	0.90	1.00	0.80	1.00	1.00	0.10
California Housing	0.40	1.00	1.00	1.00	1.00	0.50
CPU Act	0.90	1.00	0.90	1.00	1.00	1.00
CPU Small	1.00	1.00	0.80	1.00	1.00	1.00
Delta Ailerons	0.00	0.90	0.50	0.50	1.00	0.20
Delta Elevators	0.60	1.00	0.80	0.90	1.00	1.00
Diabetes	0.50	0.40	0.90	0.40	0.80	0.80
Elevators	1.00	1.00	0.20	1.00	1.00	0.40
Friedman Artificial	1.00	1.00	1.00	1.00	1.00	1.00
House 16H	1.00	1.00	0.90	1.00	0.90	0.90
House 8L	1.00	0.90	0.90	1.00	1.00	0.30
Kinematics	1.00	1.00	0.90	1.00	1.00	1.00
Machine CPU	1.00	1.00	0.80	1.00	1.00	0.70
MV Artificial	0.00	1.00	0.70	1.00	1.00	0.60
Pole Telecom	1.00	0.00	1.00	1.00	0.80	0.90
Pumadyn 32H	0.40	1.00	0.90	1.00	0.90	0.30
Pumadyn 8NH	1.00	0.40	0.40	1.00	0.30	0.60
Pyrimidines	0.90	0.90	0.40	0.90	1.00	0.60
Servo	0.60	1.00	0.50	0.90	1.00	0.80
Stocks	0.00	1.00	0.60	1.00	1.00	1.00
Triazines	0.70	0.90	0.50	0.90	0.50	0.90
Wisconsin Breast C.	0.80	1.00	0.90	0.40	1.00	0.00
Average	0.73	0.91	0.74	0.93	0.94	0.72
Significance Level	++	+++	++	+++	+++	++

Table 5: The reversed normalized rank of the true class order in comparison to random permutations ($m = 5$). A value of 1 (0) means that the true order was better (worse) than all other permutations. Under the null-hypothesis of an irrelevant class order, the expected rank is 0.5. The +, ++, +++ signs indicate, respectively, that the null-hypothesis was rejected at a significance level of $\alpha = 0.1$, $\alpha = 0.05$, $\alpha = 0.01$.

$m = 7$	EOND			FH		
Dataset	C4.5	LR	VP	C4.5	LR	VP
2D Planes	1.00	1.00	1.00	1.00	1.00	1.00
Abalone	0.50	1.00	0.90	1.00	1.00	1.00
Ailerons	0.50	1.00	0.00	1.00	1.00	0.40
Auto MPG	1.00	1.00	1.00	1.00	1.00	0.80
Auto Price	1.00	1.00	0.80	1.00	1.00	0.80
Bank 32NH	1.00	1.00	1.00	1.00	1.00	1.00
Bank 8FM	0.60	1.00	1.00	1.00	1.00	1.00
Boston Housing	1.00	1.00	0.70	1.00	1.00	0.90
California Housing	0.60	1.00	0.90	1.00	1.00	0.80
CPU Act	0.90	1.00	0.90	1.00	1.00	1.00
CPU Small	1.00	1.00	0.90	1.00	1.00	1.00
Delta Ailerons	0.20	1.00	0.50	1.00	1.00	0.50
Delta Elevators	0.10	0.90	0.60	0.90	1.00	0.70
Diabetes	0.50	0.80	1.00	1.00	0.90	0.90
Elevators	1.00	1.00	0.80	1.00	1.00	0.60
Friedman Artificial	0.90	1.00	1.00	1.00	1.00	1.00
House 16H	0.80	1.00	0.90	1.00	0.90	0.60
House 8L	1.00	1.00	1.00	1.00	1.00	0.50
Kinematics	1.00	1.00	0.90	1.00	1.00	0.80
Machine CPU	0.90	1.00	1.00	1.00	1.00	0.70
MV Artificial	0.00	1.00	0.90	1.00	1.00	0.50
Pole Telecom	0.90	0.40	0.90	1.00	0.50	0.40
Pumadyn 32H	0.30	1.00	1.00	1.00	1.00	0.80
Pumadyn 8NH	0.90	0.30	0.80	1.00	0.60	0.90
Pyrimidines	0.70	1.00	1.00	1.00	1.00	1.00
Servo	0.70	1.00	1.00	1.00	1.00	1.00
Stocks	0.60	1.00	0.90	1.00	1.00	0.60
Triazines	1.00	1.00	0.90	1.00	0.80	1.00
Wisconsin Breast C.	0.00	0.70	0.10	1.00	0.80	0.70
Average	0.71	0.93	0.84	1.00	0.95	0.79
Significance Level	++	+++	+++	+++	+++	+++

Table 6: The reversed normalized rank of the true class order in comparison to random permutations ($m = 7$). A value of 1 (0) means that the true order was better (worse) than all other permutations. Under the null-hypothesis of an irrelevant class order, the expected rank is 0.5. The +, ++, +++ signs indicate, respectively, that the null-hypothesis was rejected at a significance level of $\alpha = 0.1$, $\alpha = 0.05$, $\alpha = 0.01$.

		EOND					FH				
Dataset	m	C4.5	LR	VP	SVM-	SVM-	C4.5	LR	VP	SVM-	SVM-
					LIN	RBF				LIN	RBF
Asbestos	3	0.50	0.00	1.00	0.50	0.50	0.00	0.50	1.00	0.00	1.00
Balance Scale	3	0.00	1.00	0.00	1.00	1.00	0.00	1.00	1.00	1.00	1.00
CMC	3	0.00	1.00	0.00	1.00	1.00	0.00	1.00	0.00	1.00	0.50
Pasture Production	3	1.00	1.00	0.00	1.00	0.50	1.00	1.00	0.50	1.00	0.50
Postoperative	3	0.50	0.50	0.00	1.00	1.00	0.50	1.00	0.00	1.00	1.00
Squash (unstored)	3	0.00	0.50	1.00	0.00	0.50	0.00	0.50	1.00	0.50	0.00
Car	4	0.91	0.91	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
Grub Damage	4	0.45	1.00	0.27	1.00	0.91	0.73	0.91	0.18	1.00	0.82
Nursery 1k	4	0.00	0.36	0.91	0.64	1.00	1.00	1.00	0.82	0.73	0.18
SWD	4	0.55	0.36	0.55	0.55	0.55	1.00	0.82	1.00	0.91	0.91
Bondrate	5	0.20	0.70	0.80	0.70	0.20	0.40	1.00	0.70	1.00	0.90
Eucalyptus	5	1.00	1.00	0.30	1.00	1.00	1.00	1.00	0.10	1.00	1.00
LEV	5	0.40	1.00	0.70	1.00	1.00	1.00	1.00	1.00	1.00	1.00
ERA	9	0.70	0.90	0.00	0.80	0.30	1.00	1.00	0.90	0.90	0.40
ESL	9	0.90	0.90	0.20	0.90	0.90	1.00	1.00	0.20	1.00	1.00
Average		0.47	0.74	0.45	0.81	0.76	0.64	0.92	0.63	0.87	0.75
Significance Level					+++			+++		+++	

Table 7: The reversed normalized rank of the true class order in comparison to random permutations. A value of 1 (0) means that the true order was better (worse) than all other permutations. Under the null-hypothesis of an irrelevant class order, the expected rank is 0.5. The +, ++, +++ signs indicate, respectively, that the null-hypothesis was rejected at a significance level $\alpha = 0.1$, $\alpha = 0.05$, $\alpha = 0.01$.

m=5	C4.5			LR			VP		
Dataset	EOND	FH	RR	EOND	FH	RR	EOND	FH	RR
2D Planes	75.3	75.3	75.3	65.2	65.1	66.4	65.9	65.9	65.9
Abalone	49.5	48.4	49.2	52.2	51.4	52.4	48.5	45.7	49.4
Ailerons	58.9	57.5	57.3	61.3	61.0	61.1	24.6	23.2	26.3
Auto MPG	60.6	60.9	57.8	66.9	66.9	66.0	22.8	24.5	19.5
Auto Price	63.7	61.5	61.5	55.3	55.1	56.3	30.5	25.6	33.9
Bank 32NH	40.7	38.3	39.6	48.4	48.2	48.1	47.8	47.0	47.8
Bank 8FM	74.4	73.2	73.4	78.9	78.9	79.0	74.4	74.6	74.3
Boston Housing	61.8	60.5	61.0	64.9	64.9	64.5	39.2	31.3	39.3
California Housing	65.7	63.2	63.7	54.8	54.6	54.9	33.8	27.5	33.8
CPU Act	69.6	67.3	67.4	71.9	71.8	71.9	38.7	40.4	38.4
CPU Small	66.2	64.2	64.3	66.7	66.7	66.6	38.7	40.4	38.4
Delta Ailerons	56.2	54.7	55.9	56.4	55.7	56.8	46.7	38.6	48.1
Delta Elevators	48.9	47.7	48.7	49.4	49.3	49.4	39.7	38.4	40.5
Diabetes	26.9	24.6	25.2	25.0	26.6	25.3	29.4	28.4	26.2
Elevators	52.0	48.9	50.5	57.0	56.9	57.3	31.6	28.9	33.2
Friedman Artificial	67.6	65.6	65.4	55.5	55.4	55.4	55.2	54.7	55.5
House 16H	54.0	51.0	52.3	49.1	46.9	49.7	20.0	20.0	20.0
House 8L	54.2	52.4	52.9	49.6	47.6	50.5	20.5	20.3	20.1
Kinematics	48.9	46.2	46.6	41.6	41.7	40.5	40.8	38.4	41.1
Machine CPU	57.0	56.7	56.3	60.9	60.5	60.9	20.4	22.7	19.9
MV Artificial	99.1	99.0	99.0	89.8	88.3	91.4	49.7	31.3	57.1
Pole Telecom	94.7	94.3	94.3	82.0	80.8	82.7	81.5	80.4	80.5
Pumadyn 32H	65.9	64.7	64.7	38.5	38.9	37.8	39.1	28.2	37.8
Pumadyn 8NH	50.7	49.7	50.2	47.4	45.7	48.3	44.9	37.7	47.3
Pyrimidines	41.4	42.0	41.0	47.0	46.9	48.2	33.9	29.2	33.6
Servo	63.4	55.6	63.9	70.1	68.7	68.5	55.6	50.7	54.5
Stocks	84.9	84.5	84.1	81.5	77.8	82.2	59.4	49.9	60.8
Triazines	37.1	34.8	35.6	35.8	33.6	35.3	30.6	27.9	30.7
Wisconsin Breast C.	23.5	21.2	23.8	22.5	21.9	20.4	20.5	19.5	19.4
Average Rank (overall)	2.8	4.7	4.1	3.4	4.3	3.5	7.0	7.9	7.2
Average Rank (grouped)	1.2	2.6	2.3	1.7	2.5	1.8	1.8	2.4	1.8
Significance Level									
EOND	X	+++	+++	X	++		X		
FH	- - -	X		- -	X	- -		X	
RR	- - -		X		++	X			X

Table 8: Classification accuracies for the meta- and base-classifiers ($m = 5$). The significance of the differences between the meta-classifier is indicated at the bottom of the table.

m=7	C4.5			LR			VP		
Dataset	EOND	FH	RR	EOND	FH	RR	EOND	FH	RR
2D Planes	65.7	65.7	65.6	55.1	54.9	55.9	56.5	56.7	56.6
Abalone	37.7	36.1	37.3	40.4	40.3	40.3	35.4	32.8	36.4
Ailerons	49.7	47.9	48.2	52.0	51.3	51.7	18.8	14.8	22.2
Auto MPG	50.3	48.6	48.6	52.2	53.1	50.4	15.1	16.2	13.4
Auto Price	53.4	55.4	49.4	48.9	47.0	51.3	22.4	19.0	25.1
Bank 32NH	33.6	30.2	32.6	39.5	39.4	39.4	38.6	37.3	38.5
Bank 8FM	65.2	64.0	63.5	70.3	70.3	70.3	63.4	63.9	62.9
Boston Housing	50.4	49.4	48.6	52.2	51.5	50.8	27.5	25.0	28.8
California Housing	55.7	53.1	53.2	44.3	44.1	44.6	24.9	19.4	25.7
CPU Act	60.7	57.7	58.3	63.0	62.8	62.9	30.4	29.0	31.6
CPU Small	57.4	55.0	55.1	58.1	58.1	57.9	30.5	29.0	31.6
Delta Ailerons	47.2	45.0	46.7	47.3	46.9	47.4	36.4	29.3	37.8
Delta Elevators	43.0	40.8	42.7	43.3	43.3	43.6	34.2	31.9	35.5
Diabetes	24.3	22.1	26.1	20.0	18.1	19.5	22.8	21.5	20.7
Elevators	41.7	38.7	40.0	47.0	46.7	47.2	23.6	20.8	24.2
Friedman Artificial	57.1	54.9	54.3	44.3	44.1	44.2	43.6	42.7	43.9
House 16H	43.9	40.5	42.0	38.8	35.1	39.6	14.3	14.3	14.3
House 8L	44.0	41.4	42.9	39.6	37.5	40.7	14.9	14.8	14.3
Kinematics	38.2	35.0	36.2	31.9	32.2	31.1	30.6	26.9	31.4
Machine CPU	45.5	45.1	44.0	49.9	49.9	49.3	15.6	15.7	14.5
MV Artificial	98.7	98.6	98.6	90.5	87.7	92.0	39.7	18.1	42.3
Pole Telecom	93.4	92.9	92.6	81.6	80.6	82.2	81.2	78.3	80.4
Pumadyn 32H	56.0	54.4	54.9	30.5	30.7	29.4	30.4	18.5	29.3
Pumadyn 8NH	38.9	36.8	38.7	35.5	34.3	36.2	34.6	27.8	35.5
Pyrimidines	27.9	30.7	25.4	33.9	32.7	32.6	25.5	22.6	21.4
Servo	44.9	38.0	46.1	51.2	50.4	48.7	40.9	37.2	40.2
Stocks	80.7	79.1	78.8	73.9	68.6	74.8	47.1	32.4	52.1
Triazines	31.0	26.7	27.1	27.7	24.8	27.3	23.8	22.4	23.3
Wisconsin Breast C.	15.0	15.3	16.8	14.0	15.0	13.5	13.7	14.7	14.2
Average Rank (overall)	2.8	4.2	4.0	3.4	4.4	3.7	7.2	8.1	7.2
Average Rank (grouped)	1.2	2.5	2.3	1.6	2.5	1.9	1.8	2.5	1.7
Significance Level									
EOND	X	+++	++	X	++		X		
FH	- - -	X		- -	X			X	-
RR	- -		X			X		+	X

Table 9: Classification accuracies for the meta- and base-classifiers ($m = 7$). The significance of the differences between the meta-classifier is indicated at the bottom of the table.

		C4.5			LR			VP			SVM-LIN			SVM-RBF		
Dataset	m	EOND	FH	RR	EOND	FH	RR	EOND	FH	RR	EOND	FH	RR	EOND	FH	RR
Asbestos	3	72.0	70.2	71.7	73.6	74.1	73.4	54.2	54.2	54.2	73.1	72.1	73.2	75.7	76.3	74.8
Balance Scale	3	79.4	75.9	79.3	89.5	89.7	89.5	87.3	89.7	87.5	90.8	91.2	90.9	89.7	88.9	90.1
CMC	3	51.8	48.1	52.2	50.8	50.8	50.7	42.6	33.9	42.4	49.9	49.9	49.2	48.7	45.7	47.3
Pasture P.	3	73.1	74.0	70.0	67.1	61.4	68.9	36.1	36.4	38.6	69.0	71.0	69.0	73.4	72.4	74.4
Postoperative	3	70.0	70.0	70.1	59.0	62.3	58.7	66.7	65.2	67.3	62.8	64.0	62.5	57.4	58.3	56.8
Squash (unst.)	3	74.8	63.2	76.7	64.0	57.0	72.3	53.4	52.0	50.3	72.0	64.3	73.9	70.7	64.6	72.9
Car	4	92.4	90.2	92.9	93.4	93.1	93.8	90.1	89.4	89.5	92.8	92.1	93.1	90.7	91.1	89.7
Grub Damage	4	42.9	38.6	43.3	40.2	40.2	37.0	38.6	31.0	39.4	45.4	44.1	43.7	46.8	44.6	46.1
Nursery 1k	4	88.1	88.7	88.0	92.3	92.4	92.2	90.7	89.8	90.2	92.2	91.9	92.3	91.4	89.6	91.5
SWD	4	56.9	56.8	56.5	57.3	57.1	57.5	48.5	48.4	48.1	57.4	57.2	57.3	57.7	57.4	57.5
Bondrate	5	54.4	51.5	54.9	41.5	42.3	43.0	50.7	44.3	49.1	53.7	54.1	53.5	49.0	51.3	50.5
Eucalyptus	5	63.1	62.4	61.6	65.1	64.8	63.0	30.5	18.2	33.4	65.3	64.5	64.9	57.8	57.4	50.8
LEV	5	60.2	60.4	59.5	60.3	60.2	59.7	50.1	51.0	49.5	59.4	59.0	59.1	59.8	59.5	59.0
ERA	9	56.2	56.4	55.6	56.0	56.0	56.2	41.6	42.2	43.1	55.1	54.7	55.6	54.9	53.9	55.3
ESL	9	83.7	83.9	83.4	87.9	88.0	87.8	51.6	48.1	52.7	88.2	88.2	87.9	87.0	86.3	86.9
Avg. Rank (ov.)		6.5	8.4	7.3	6.4	6.7	6.9	12.1	12.7	12.3	5.4	6.6	5.7	6.9	8.3	7.3
Avg. Rank (gr.)		1.8	2.2	2.1	1.9	1.9	2.2	1.8	2.4	1.8	1.7	2.3	2.1	1.7	2.3	2.0
Sign. Level																
EOND		X			X			X			X			X		
FH			X			X			X			X			X	
RR				X			X			X			X			X

Table 10: Classification accuracies for the meta- and base-classifiers. The significance of the differences between the meta-classifier is indicated at the bottom of the table.

References

A. Asuncion and D.J. Newman. UCI machine learning repository. http://www.ics.uci.edu/∼mlearn/MLRepository.html, 2007.

R. Babaria, J. S. Nath, S. Krishnan, K. R. Sivaramakrishnan, C. Bhattacharyya, and M. N. Murty. Focused crawling with scalable ordinal regression solvers. In *ICML–07, Proceedings of the 24th International Conference on Machine Learning*, pages 57–64, New York, NY, USA, 2007.

D. Barker. Dataset: Pasture production. http://weka.sourceforge.net/wiki/index.php/Datasets, 2007. Obtained on 20th of October 2007.

B. Bulloch. Dataset: Eucalyptus soil conservation. http://weka.sourceforge.net/wiki/index.php/Datasets, 2007. Obtained on 20th of October 2007.

J. S. Cardoso and J. F. Pinto da Costa. Learning to classify ordinal data: The data replication method. *The Journal of Machine Learning Research*, 8:1393–1429, 2007.

J. S. Cardoso, J. F. Pinto da Costa, and M. J. Cardoso. Modelling ordinal relations with SVMs: an application to objective aesthetic evaluation of breast cancer conservative treatment. *Neural Networks*, 18(5-6):808–817, 2005.

W. Chu and S. S. Keerthi. New approaches to support vector ordinal regression. In *ICML–05, Proceedings of the 22nd International Conference on Machine Learning*, pages 145–152, New York, NY, USA, 2005.

A. Ben David. Dataset: Ordinal real-world datasets (esl, era, lev, swd). http://weka.sourceforge.net/wiki/index.php/Datasets, 2008. Obtained on 15th of March 2008.

J. Demšar. Statistical comparisons of classifiers over multiple data sets. *The Journal of Machine Learning Research*, 7: 1–30, 2006.

E. Frank and M. Hall. A simple approach to ordinal classification. In *EMCL–01, Proceedings of the 12th European Conference on Machine Learning*, pages 145–156, London, UK, 2001.

E. Frank and S. Kramer. Ensembles of nested dichotomies for multi-class problems. In *ICML–04, Proceedings of the 21st International Conference on Machine Learning*, pages 305–312, New York, NY, USA, 2004.

Y. Freund and R. E. Schapire. Large margin classification using the perceptron algorithm. *Machine Learning*, 37(3): 277–296, 1999.

M. Friedman. The use of ranks to avoid the assumption of normality implicit in the analysis of variance. *Journal of the American Statistical Association*, 32(200):675–701, 1937.

M. Friedman. A comparison of alternative tests of significance for the problem of m rankings. *The Annals of Mathematical Statistics*, 11(1):86–92, 1940.

J. Fürnkranz. Round robin classification. *The Journal of Machine Learning Research*, 2:721–747, 2002a.

J. Fürnkranz. Pairwise classification as an ensemble technique. In *ECML–02, Proceedings of the 13th European Conference on Machine Learning*, pages 97–110, London, UK, 2002b.

W. Harvey. Dataset: Squash harvest stored / unstored. http://weka.sourceforge.net/wiki/index.php/Datasets, 2007. Obtained on 20th of October 2007.

M. Meyer and P. Vlachos. Statlib. http://lib.stat.cmu.edu/, 2007.

P. Nemenyi. *Distribution-free multiple comparisons.* PhD thesis, Princeton University, 1963.

J. Platt. Probabilistic outputs for support vector machines and comparison to regularized likelihood methods. In A.J. Smola, P. Bartlett, B. Schoelkopf, and D. Schuurmans, editors, *Advances in Large Margin Classifiers*, pages 61–74, Cambridge, MA, 1999. MIT Press.

J. R. Quinlan. *C4.5: Programs for Machine Learning.* Morgan Kaufmann, 1993.

R. J. Townsend. Dataset: Grass grubs and damage ranking. http://weka.sourceforge.net/wiki/index.php/Datasets, 2007. Obtained on 20th of October 2007.

V. N. Vapnik. *The Nature of Statistical Learning Theory.* Springer-Verlag, 1999.

$m = 5$	EOND-4			RR			EOND-1			FH		
Dataset	C4.5	LR	VP	C4.5	LR	VP	C4.5	LR	VP	C4.5	LR	VP
2D Planes	75.3	65.1	65.7	75.3	66.5	65.9	75.3	65.0	65.4	75.3	65.1	66.0
Abalone	49.2	52.2	47.7	49.2	52.4	49.4	48.5	52.0	47.0	48.4	51.4	45.7
Ailerons	58.0	61.2	24.1	57.3	61.2	26.3	56.8	61.2	23.9	57.5	61.0	23.2
Auto MPG	59.9	66.8	21.7	57.8	66.0	19.5	58.3	66.8	22.8	60.7	66.9	24.6
Auto Price	62.5	54.6	27.9	61.5	56.3	33.9	61.7	53.6	26.2	61.5	55.1	25.6
Bank 32NH	39.3	48.3	47.4	39.6	48.2	47.8	38.0	48.2	47.1	38.4	48.2	47.0
Bank 8FM	73.7	78.9	74.4	73.4	79.0	74.3	73.0	78.9	74.4	73.2	78.9	74.6
Boston Hous.	61.0	65.1	36.4	61.0	64.5	39.3	59.7	64.9	34.9	60.7	65.0	31.3
Cal. Hous.	64.8	54.7	31.9	63.7	54.9	33.8	63.1	54.7	30.7	63.2	54.6	27.5
CPU Act	68.1	71.9	39.0	67.4	71.9	38.4	67.0	71.8	38.6	67.3	71.8	40.4
CPU Small	65.1	66.7	39.6	64.3	66.7	38.4	64.0	66.6	39.0	64.3	66.7	40.4
D. Ailerons	56.0	56.4	46.2	55.9	56.8	48.1	55.5	56.3	45.5	54.7	55.7	38.7
D. Elevators	48.6	49.4	39.2	48.7	49.4	40.5	48.3	49.4	39.0	47.7	49.3	38.4
Diabetes	27.1	24.9	28.5	25.2	25.3	26.2	27.0	25.5	25.5	24.5	26.4	28.3
Elevators	50.9	57.1	31.5	50.5	57.3	33.2	49.1	57.0	31.0	48.9	57.0	28.9
Friedman Art.	66.5	55.5	54.7	65.4	55.5	55.5	65.1	55.4	54.8	65.6	55.4	54.7
House 16H	52.8	49.0	20.0	52.3	49.7	20.0	51.3	48.6	20.0	51.0	46.9	20.0
House 8L	53.6	49.5	20.7	52.9	50.5	20.1	52.5	49.0	20.8	52.5	47.6	20.3
Kinematics	47.7	41.6	40.0	46.6	40.6	41.1	45.0	41.5	39.2	46.2	41.7	38.4
Machine CPU	56.3	60.8	20.9	56.4	60.9	19.9	56.0	60.6	21.7	56.8	60.5	22.7
MV Artificial	99.1	89.6	44.8	99.0	91.4	57.1	99.0	89.5	41.3	99.0	88.3	31.3
Pole Telecom	94.4	81.9	80.9	94.3	82.8	80.5	94.2	81.7	80.2	94.3	80.8	80.4
Pumadyn 32H	64.9	38.5	36.8	64.7	37.8	37.8	64.3	38.5	33.9	64.7	38.9	28.3
Pumadyn 8NH	50.5	47.4	43.1	50.2	48.3	47.3	50.0	47.3	42.0	49.7	45.7	37.7
Pyrimidines	41.3	46.4	32.6	41.0	48.2	33.6	39.9	45.0	30.6	42.1	46.8	29.2
Servo	61.4	69.3	53.6	63.9	68.6	54.5	60.7	68.7	52.7	55.5	68.7	50.5
Stocks	84.5	81.2	59.0	84.2	82.2	60.8	84.0	79.7	56.1	84.5	77.9	49.9
Triazines	36.3	34.9	29.8	35.6	35.3	30.7	35.7	33.4	28.2	34.9	33.6	27.9
W. Breast C.	21.5	21.9	20.0	23.8	20.5	19.4	22.3	22.4	20.0	21.1	22.1	19.5

Table 11: Classification accuracies for variants of EOND in comparison to RR and FH.

I. H. Witten and E. Frank. *Data Mining: Practical Machine Learning Tools and Techniques.* Morgan Kaufmann, San Francisco, 2nd edition, 2005.

S. Yu, K. Yu, V. Tresp, and H.-P. Kriegel. Collaborative ordinal regression. In *ICML–06, Proceedings of the 23rd International Conference on Machine Learning*, pages 1089–1096, New York, NY, USA, 2006. ACM.

Biographical Notes

Jens Christian Hühn received the MSc Degree in Computer Science from the University of Marburg (Germany) in May 2006. Since then, he is a PhD student in the Knowledge Engineering and Bioinformatics Lab at the same university. Currently, he is working on different topics in machine learning, inlcuding fuzzy rule learning, reliable classification, and ordinal classification. He holds a scholarship from the German Konrad-Adenauer-Foundation.

Eyke Hüllermeier is a Full Professor in the Department of Mathematics and Computer Science at Marburg University (Germany), where he heads the Knowledge Engineering and Bioinformatics Lab. He holds a PhD in Computer Science (1997) and a habilitation degree (2002), both from the University of Paderborn (Germany). His research interests are focused on machine learning and data mining, including both methodical foundations and applications in different fields, especially bioinformatics. He is on the editorial board of the journals Fuzzy Sets and Systems, Soft Computing, Advances in Fuzzy Systems, and the International Journal of Data Mining, Modelling and Management.