FR3: A Fuzzy Rule Learner for Inducing Reliable Classifiers

Jens Hühn and Eyke Hüllermeier
Philipps-Universität Marburg
Department of Mathematics and Computer Science

Abstract

This paper introduces a fuzzy rule-based classification method called FR3, which is short for Fuzzy Round Robin RIPPER. In the context of polychotomous classification, it uses a fuzzy extension of the well-known RIPPER algorithm as a base learner within a round robin scheme. A key feature of FR3 is its ability to represent different facets of uncertainty involved in a classification decision in a more faithful way, thereby providing the basis for implementing "reliable classifiers" that may, for example, abstain from a decision when not being sure enough.

1 Introduction

A close connection between classification learning, on the one side, and fuzzy preference modeling and decision making, on the other side, has recently been established by Hüllermeier and Brinker in [HB08]. The idea of their approach is to reduce a problem of polychotomous classification, involving m classes $\mathcal{L} = \{\lambda_1 \ldots \lambda_m\}$, to a problem of decision making based on a fuzzy preference structure. Following a round robin scheme, their approach, called LVPC (Learning Valued Preferences for Classification), first trains an ensemble of binary models, one for every pair of classes. Then, given a query instance $\boldsymbol{x}$ to be classified, three fuzzy relations (in the

form of $\{1 \ldots m\} \times \{1 \ldots m\} \to [0,1]$ mappings) can be derived from the predictions of this ensemble. For every pair of labels (λ_i, λ_j), the corresponding entries in these relations express, respectively, a degree of

- **preference:** the degree to which the label λ_i is (strictly) preferred to λ_j as a classification for $\boldsymbol{x}$ (and vice versa);

- **conflict:** the degree to which λ_i and λ_j are in conflict with each other (as both of them are supported simultaneously as potential classifications);

- **ignorance:** the degree of ignorance reflecting to what extent neither λ_i nor λ_j is supported as a classification.

A final classification, or any other type of decision (e.g., to abstain or to gather additional information), can then be made on the basis of these relations.

A key feature of this approach is its ability to represent ignorance in a faithful way. In fact, even though many machine learning methods are able to reflect conflict in one way or the other, for example in terms of probability distributions, the same is not true for ignorance. To illustrate the meaning of conflict and ignorance in the context of classification, consider the simple scenario shown in Fig. 1: Given observations from two classes, `black` and `white`, three new instances marked by a cross need to be classified. Obviously, given the current observations, the upper left instance can quite safely be classified as `white`. The case of the lower left instance, however, involves a
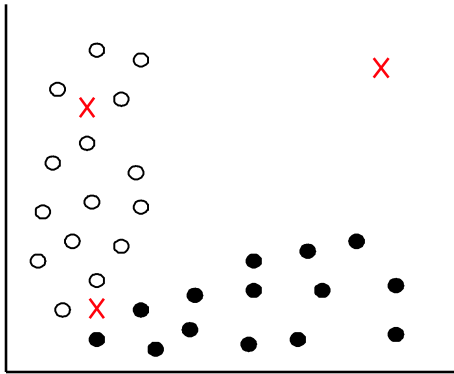
Figure 1: Classification scenario: Observations from two classes (points) and new query instances (crosses).
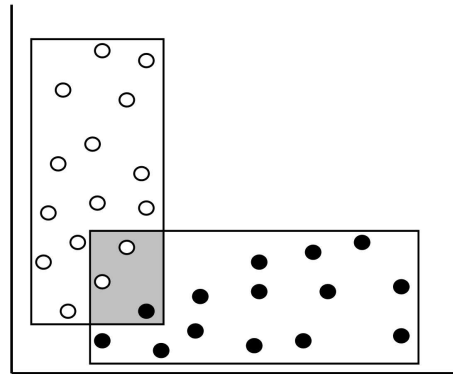

Figure 2: Regions of conflict (gray area) and ignorance (area not covered by any rule) in case of a rule-based model (rules indicated as rectangles).

high level of conflict, since both classes, `black` and `white`, appear plausible. The third situation is an example of ignorance: The upper right instance is located in a region of the instance space in which no observations have been made so far. Consequently, there is neither evidence in favor of class `black` nor in favor of class `white`.

It was already mentioned in [HB08] that rule-based classifiers are, in principle, ideally suited for implementing the pairwise models needed in LVPC. The main reason for this suitability is that, in contrast to standard discriminative classification methods (such as linear discriminant functions), rule-based models are able to represent conflict and, more importantly, ignorance in a natural way: A situation of conflict occurs if an instance x is simultaneously covered by two (or more) conflicting rules, while a situation of ignorance occurs if it is not covered by any rule; see Fig. 2.

In this regard, however, conventional rule-based classifiers can be criticized for at least two reasons: First, many approaches induce proper rules only for one class, typically the minority class, and add a default rule that predicts the other class in case no other rule applies. Thus, ignorance is eliminated in an artificial and arguably questionable way. In fact, note that this approach may come along with a high level of *extrapolation*, since the default class can be predicted in regions where

it has never been observed before.

Second, since conventional (non-fuzzy) rules have "sharp boundaries", they produce an abrupt transition between support of a class and ignorance which is not very natural. Intuitively, the farther away an instance is located from the core of the closest rule, the higher the degree of ignorance should be. Or, stated differently, the support provided by a rule should decrease from "full" (inside the core) to "zero" in a gradual instead of an abrupt way.

To address these two issues, we propose to use fuzzy rules instead of conventional rules. More specifically, we develop a fuzzy extension of RIPPER [Coh95], a state-of-the-art rule induction algorithm that produces accurate models in an efficient way. By using the fuzzy instead of the original version of RIPPER as a base learner within the round robin (all-pairs) decomposition scheme, we extend the R3 method proposed by Fürnkranz in [Für02]. Experimentally, it will be shown that our approach, called FR3, is not only able to reflect conflict and ignorance of a classification in a faithful way, but also outperforms R3 in terms of predictive accuracy.

2 Round Robin RIPPER

Consider a polychotomous classification problem involving m classes $\mathbb{L} \stackrel{\mathrm{df}}{=} \{\lambda_1 \ldots \lambda_m\}$. Suppose instances to be represented in terms of

attributes A_i, $i = 1 \ldots n$, which are either numerical (real-valued) or nominal, and let $\mathbb{D}_i$ denote the corresponding domains. Thus, an instance is represented as an n-dimensional attribute vector

$$\boldsymbol{x} = (x_1 \ldots x_n) \in \mathbb{D} \stackrel{\text{df}}{=} \mathbb{D}_1 \times \ldots \times \mathbb{D}_n.$$

RIPPER is a state-of-the-art rule learning algorithm that was introduced in [Coh95] as a successor of the IREP algorithm for rule induction [FW94]. A single RIPPER rule is of the form $r = \langle r_A \,|\, r_C \rangle$, consisting of an antecedent part r_A and a consequent part r_C. The antecedent part r_A is a conjunction of predicates (selectors) which are of the form $(A_i = v)$ for nominal and $(A_i \,\theta\, v)$ for numerical attributes, where $\theta \in \{\leq, =, \geq\}$ and $v_i \in \mathbb{D}_i$. The consequent part r_C is a class assignment of the form (class $= \lambda$), where $\lambda \in \mathbb{L}$. A rule $r = \langle r_A \,|\, r_C \rangle$ is said to *cover* an instance $\boldsymbol{x} = (x_1 \ldots x_n)$ if the attribute values x_i satisfy all the predicates in r_A.

Due to reasons of space we refrain from a detailed review of the RIPPER rule induction algorithm and instead refer to the extended version of this paper [HH08].

Round robin learning *aka* all-pairs or all-versus-all learning is a special decomposition technique that transforms a multi-class classification problem involving $m > 2$ classes $\mathbb{L} = \{\lambda_1 \ldots \lambda_m\}$ into a number of *binary* problems. To this end, a separate model (base learner) $\mathcal{M}_{i,j}$ is trained for each *pair* of labels (λ_i, λ_j), $1 \leq i < j \leq m$. $\mathcal{M}_{i,j}$ is intended to separate the objects with label λ_i from those having label λ_j. If $(\boldsymbol{x}, \lambda_a) \in \mathbb{D} \times \mathbb{L}$ is an original training example (revealing that instance $\boldsymbol{x}$ has label λ_a), then $\boldsymbol{x}$ is considered as a *positive* example for all learners $\mathcal{M}_{a,j}$, $j < a$, and as a *negative* example for the learners $\mathcal{M}_{i,a}$, $i < a$.

At classification time, a query $\boldsymbol{x}$ is submitted to all $m(m-1)/2$ learners, and each prediction $\mathcal{M}_{i,j}(\boldsymbol{x})$ is typically interpreted as a vote for a label. Assuming models in the form of $[0, 1]$-valued (scoring) classifiers, an output close to 1 indicates support of λ_i, whereas an output close to 0 is counted as evidence in favor of λ_j. The simplest classification strategy, then,

is to predict the class label with the highest score in terms of the sum of (weighted) votes:

$$s_i \stackrel{\text{df}}{=} \sum_{1 \leq j \neq i \leq m} s_{i,j}, \tag{1}$$

where $s_{i,j} = \mathcal{M}_{i,j}(\boldsymbol{x})$ for $i < j$ and $s_{i,j} = 1 - \mathcal{M}_{j,i}(\boldsymbol{x})$ for $j < i$.

Even though the main purpose of decomposition techniques is to enable the application of methods that are inherently limited to binary classification, such as support vector machines, to polychotomous problems, round robin learning can be interesting even in the case where the models $\mathcal{M}$ can, in principle, handle multi-class problems in a direct way. The main reason is that the binary problems are often much simpler than the original m-class problem, so that models induced from data become more accurate and more stable. In particular, for the case of RIPPER, Fürnkranz [Für02, Für03] showed that a Round Robin RIPPER (R3), i.e., an all-pairs classifier with RIPPER as a (binary) base learner, outperforms the original multi-class RIPPER.

Apart from that, the all-pairs decomposition technique is essential for the LVPC approach proposed in [HB08], namely for producing the (binary) relations that constitute a fuzzy preference structure (cf. Section 3.5).

3 Fuzzy Round Robin RIPPER

In this section, we introduce the Fuzzy Round Robin RIPPER (FR3) approach. To this end, we first propose a fuzzy version of the basic RIPPER, called FRIPPER. In a second step, FRIPPER is then integrated as a base learner in a round robin learning scheme. FRIPPER modifies the original RIPPER algorithm in several ways, as will be seen in the following subsections, in which we focus on the two-class case. The multi-class case, in connection with round robin learning, will then be addressed in the final subsection. Due to space restrictions, we restrict ourselves to highlighting the main points and again refer to [HH08] for a more detailed exposition.

3.1 Learning Rules for Both Classes

A first modification of RIPPER concerns its use of default rules. As mentioned previously, using one class as a default prediction is disadvantageous with regard to reliable classification and, in particular, hinders a faithful representation of ignorance. Besides, this strategy comes along with a systematic bias in favor of those classes chosen as a default.

To represent ignorance, a classifier must be able to abstain, that is, to refrain from supporting any class. To achieve this, we train two classifiers for every pairwise model, and combine them by merging the respective proper rules, i.e., the non-default rules.

3.2 Representation of Fuzzy Rules

A selector constraining a numerical attribute A_i (with domain $\mathbb{D}_i = \mathbb{R}$) in a RIPPER rule can obviously be expressed in the form $(A_i \in I)$, where $I \subseteq \mathbb{R}$ is an interval: $I = (-\infty, v]$ if the rule contains a selector $(A_i \leq v)$, $I = [u, \infty)$ if it contains a selector $(A_i \geq u)$, and $I = [u, v]$ if it contains both (in the last case, two selectors are combined).

Essentially, a fuzzy rule is obtained through replacing intervals by fuzzy intervals, namely fuzzy sets with trapezoidal membership function. A fuzzy interval of that kind is specified by four parameters and will be written $I^F = (\phi^{s,L}, \phi^{c,L}, \phi^{c,U}, \phi^{s,U})$: $\phi^{c,L}$ and $\phi^{c,U}$ are, respectively, the lower and upper bound of the core of the fuzzy set; likewise, $\phi^{s,L}$ and $\phi^{s,U}$ are, respectively, the lower and upper bound of the support. Note that, as in the non-fuzzy case, a fuzzy interval can be open to one side ($\phi^{s,L} = \phi^{c,L} = -\infty$ or $\phi^{c,U} = \phi^{s,U} = \infty$.) In fact, as will be seen later on, the fuzzy antecendents successively learned by FRIPPER are fuzzy half-intervals of exactly that kind.

A fuzzy selector $(A_i \in I_i^F)$ *covers* an instance $\boldsymbol{x} = (x_1 \ldots x_n)$ to the degree $I_i^F(x_i)$. A fuzzy rule r^F involving k selectors $(A_i \in I_i^F)$, $i = 1 \ldots k$, covers $\boldsymbol{x}$ to the degree $\mu_{r^F}(\boldsymbol{x}) = \min_{i=1\ldots k} I_i^F(x_i)$.

3.3 Rule Fuzzification

To obtain fuzzy rules, the idea is to fuzzify the final rules from our modified RIPPER algorithm. More specifically, using the training set $\mathbb{D}_T \subseteq \mathbb{D}$ for evaluating candidates, the idea is to search for the best fuzzy extension of each rule, where a fuzzy extension is understood as a rule of the same structure, but with intervals replaced by fuzzy intervals. Taking the intervals I_i of the original rules as the cores $[\phi_i^{c,L}, \phi_i^{c,U}]$ of the sought fuzzy intervals I_i^F, the problem is to find optimal bounds for the respective supports, i.e., to determine $\phi_i^{s,L}$ and $\phi_i^{s,U}$.

For the fuzzification of a single antecedent $(A_i \in I_i)$ it is important to consider only the relevant training data $D_T^{(i)}$, i.e., to ignore those instances that are excluded by any other antecedent $(A_j \in I_j^F)$, $j \neq i$:

$$\mathbb{D}_T^{(i)} = \{\boldsymbol{x} = (x_1 \ldots x_k) \in \mathbb{D}_T \mid \qquad (2)$$
$$I_j^F(x_j) > 0 \text{ for all } j \neq i\} \subseteq \mathbb{D}_T$$

We partition $\mathbb{D}_T^{(i)}$ into the subset of positive instances, $D_{T+}^{(i)}$, and negative instances, $D_{T-}^{(i)}$. To measure the quality of a fuzzification, the rule purity will be used:

$$\text{pur} = \frac{p_i}{p_i + n_i}, \qquad (3)$$

where

$$p_i \stackrel{\text{df}}{=} \sum_{\boldsymbol{x} \in D_{T+}^{(i)}} \mu_{A_i}(\boldsymbol{x})$$
$$n_i \stackrel{\text{df}}{=} \sum_{\boldsymbol{x} \in D_{T-}^{(i)}} \mu_{A_i}(\boldsymbol{x})$$

Rules are fuzzified in a greedy way. In each iteration, a fuzzification is computed for every antecedent, namely the best fuzzification in terms of (3). This is done by testing all values

$$\{x_i \mid \boldsymbol{x} = (x_1 \ldots x_k) \in D_T^{(i)}, \, x_i < \phi_i^{c,L}\}$$

as candidates for $\phi_i^{s,L}$ and, likewise, all values

$$\{x_i \mid \boldsymbol{x} = (x_1 \ldots x_k) \in D_T^{(i)}, \, x_i > \phi_i^{c,U}\}$$

as candidates for $\phi_i^{s,U}$. Ties are broken in favor of larger fuzzy sets, that is, larger distances from the core.

The fuzzification is then realized for the antecedent with the largest purity. This is repeated until all antecedent have been fuzzified.

3.3.1 Bounding Fuzzy Rules

Some fuzzy intervals may still be open to one side, which means that the corresponding rule has unbounded support. As this is not in agreement with our "cautious" extrapolation strategy, we finally close such intervals: If $\phi_j^{c,L} = -\infty$, this core bound is set to $\phi_j^{c,L} =$

$$\min\{x_j \mid \boldsymbol{x} = (x_1 \ldots x_k) \in D_{\mathrm{T}_+}, \, \mu_{I_j^F}(\boldsymbol{x}) > 0\},$$

where D_{T_+} is the subset of positive instances in D_{T}. Moreover, the support bound $\phi_j^{s,L}$ is set to the minimal value in $\mathbb{D}_j$, the domain of attribute A_j.[1] This way, the core of the rule is restricted to the region in which positive examples have indeed been observed, while the support decreases as a linear function of the distance from this region. Analogous modifications are made in the case where $\phi_j^{c,U} = \infty$.

3.4 Classifier Output

Suppose that fuzzy rules $r_1^0 \ldots r_k^0$ and $r_1^1 \ldots r_\ell^1$ have been learned, respectively, for classes λ_0 and λ_1. For a new query instance $\boldsymbol{x}$, the supports of these classes are then given, respectively, by

$$\begin{aligned} s_0 &\stackrel{\mathrm{df}}{=} \max_{i=1\ldots k} \mu_{r_i^0}(\boldsymbol{x}) \\ s_1 &\stackrel{\mathrm{df}}{=} \max_{j=1\ldots\ell} \mu_{r_j^1}(\boldsymbol{x}) \end{aligned} \quad (4)$$

From these two support degrees, the following values are derived, which constitute the output of the FRIPPER algorithm (in the two-class case):

$$\begin{aligned} P(\lambda_0, \lambda_1) &= s_0 - \min\{s_0, s_1\} \\ P(\lambda_1, \lambda_0) &= s_1 - \min\{s_0, s_1\} \\ C(\lambda_0, \lambda_1) &= \min\{s_0, s_1\} \\ I(\lambda_0, \lambda_1) &= 1 - \max\{s_0, s_1\} \end{aligned} \quad (5)$$

---

[1]More specifically, to avoid 0-memberships, we go beyond this point, as a rule of thumb by 50% of the width of $\mathbb{D}_j$.

$C(\lambda_0, \lambda_1)$ is the degree of *conflict*, namely the degree to which both classes are supported. Likewise, $I(\lambda_0, \lambda_1)$ is the degree of *ignorance*, namely the degree to which none of the classes is supported. Finally, $P(\lambda_0, \lambda_1)$ and $P(\lambda_1, \lambda_0)$ denote, respectively, the strict preference for λ_0 and λ_1. Note that at least one of these two degrees is zero, and that $P(\lambda_0, \lambda_1) + P(\lambda_1, \lambda_0) + C(\lambda_0, \lambda_1) + I(\lambda_0, \lambda_1) \equiv 1$. In passing, we also remark that (5) is actually a standard decomposition scheme, which is used in fuzzy preference modeling [FR94] to decompose a weak preference relation (here given by the support degrees s_0, s_1) into three parts: strict preference, indifference (which here corresponds to conflict), and indistinguishablity (here ignorance).

In the above approach, the purity of a fuzzy rule r has not been taken into consideration. In fact, according to (4), the support $\mu_{r_i^0}(\boldsymbol{x})$ is completely given to the class c_0, even if the purity of this rule, $\mathrm{pur}(r_i^0)$, is smaller than 1. Recalling that the purity measure can be interpreted as a kind of conditional probability (of a class given that the instance is covered by the rule), is appears more reasonable to distribute the support to λ_0 and λ_1 in a proportional way. This leads to the following modification of (4):

$$\begin{aligned} s_0 &\stackrel{\mathrm{df}}{=} \max\{\max_{i=1\ldots k} \mu_{r_i^0}(\boldsymbol{x}), \\ &\qquad \max_{j=1\ldots\ell} \mu_{r_j^1}(\boldsymbol{x})(1 - \mathrm{pur}(r_j^1))\} \\ s_1 &\stackrel{\mathrm{df}}{=} \max\{\max_{j=1\ldots\ell} \mu_{r_j^1}(\boldsymbol{x}), \\ &\qquad \max_{i=1\ldots k} \mu_{r_i^0}(\boldsymbol{x})(1 - \mathrm{pur}(r_i^0))\} \end{aligned} \quad (6)$$

Note that, by deriving (5) on the basis of (6) instead of (4), the non-purity of a rule is treated as a conflict, which makes perfect sense.

3.5 Round Robin Learning

Given a set of classes $\mathbb{L} = \{\lambda_1 \ldots \lambda_m\}$, the FRIPPER algorithm as outlined above can be applied to each pair of labels $(\lambda_k, \lambda_\ell)$, thereby producing an ensemble of models $\mathcal{M}_{k\ell}$, $1 \leq k < \ell \leq m$. A query instance $\boldsymbol{x} \in \mathbb{D}$ is then submitted to each model. As explained in Section 3.4, the output of model $\mathcal{M}_{k\ell}$ is a quadruple $\mathcal{M}_{k\ell}(\boldsymbol{x}) =$

$(p_{k\ell}, p_{\ell k}, c_{k\ell}, i_{k\ell})$, where $p_{k\ell}$ is the preference for λ_k in comparison with λ_ℓ, $p_{\ell k}$ the preference for λ_ℓ, $c_{k\ell}$ the corresponding degree of conflict, and $i_{k\ell}$ the degree of ignorance.

Thus, three relations (P, C, I) are obtained, a strict preference relation $P = (p_{k\ell})$, a conflict relation $C = (c_{k\ell})$, and an ignorance relation $I = (i_{k\ell})$; note that C and I are symmetric, so the entries in the relations are well-defined for all $1 \leq k \neq \ell \leq m$. These relations provide the basis for sophisticated classification and decision policies. For example, in the standard scenario where a single prediction is sought, the following classification rule could be used: $\lambda^* =$

$$\arg\max_{\lambda_k \in \mathbb{L}} \sum_{1 \leq \ell \neq k \leq m} p_{k\ell} + \frac{c_{k\ell}}{2} + \frac{N_k \cdot i_{k\ell}}{N_k + N_\ell}, \quad (7)$$

where N_k is the number of examples from class λ_k in the training data (and hence an unbiased estimate of the class probability). This decision rule, that turned out to perform well in practice (cf. Section 4), evaluates each candidate label in terms of the sum of strict preferences over all other labels, distributes the corresponding degrees of conflict in a uniform way and the degrees of ignorance in proportion to the size of the classes (in other words, prior probabilities are used in the case of no further information).

Going beyond the conventional classification setting, a preference structure (P, C, I) can be especially useful in generalized settings in which, for example, more than one class can be predicted in cases of conflict, or a classification decision can be refused in cases of ignorance (cf. Section 4.2).

4 Experimental Results

To analyze the performance of our FR3 approach, we conducted several experimental studies under the WEKA 3.5.5 framework [WF05]. As a starting point, we used the RIPPER implementation of WEKA ("JRip"), both for re-implementing Fürnkranz's R3 and our FR3. Besides, JRip served as a baseline in the experiments.

4.1 Classification Accuracy

In a first study, we compared RIPPER, R3, and FR3 with respect to classification accuracy, using 24 datasets from the UCI [AN07] and the STATLIB [MV07] repositories and from [Bul07, Bar07, Har07] (analcatdata-authorship, draft, halloffame, votesurvey, cars, collins, eucalyptus, glass, iris, mfeat-factors, fourier, karhunen, morphological, zernike, optdigits, page-blocks, pasture-production, pendigits, segment, squash-unstored, synthetic control, vehicle, vowel, waveform); 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. This procedure was repeated $1,000$ times to stabilize the results.

For reasons of space, we only give a summary of the results in terms of win/loss statistics and again refer to [HH08] for details. As can be seen in Table 1, the results are clearly in favor of FR3, which is significantly better (at the significance level 0.95) than the other methods.

	FR3	R3	RIPPER
FR3		20/4	24/0
R3			23/1

Table 1: Pairwise comparison between classifiers in terms of wins and losses.

4.2 Representation of Uncertainty

The ability to represent uncertainty involved in a classification decision, in terms of measures of conflict and ignorance, is arguably one of the main advantages of FR3. To test whether FR3 does indeed provide a basis for implementing classifier that are more "reliable", we conducted another series of experiments in a setting of classification with reject option. Roughly speaking, the idea is that, if γ is a reliable index of classification uncertainty, then the value of γ should correlate with the probability to make a correct decision. Or, stated differently, when abstaining from the classification of all instances the γ-value of which exceeds a threshold t, the clas-

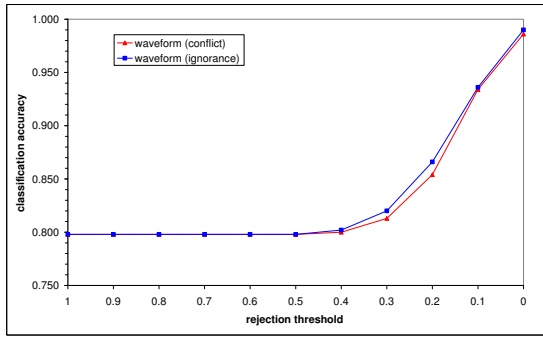Figure 3: Accuracy-rejection curves for dataset waveform.

Figure 4: Accuracy-rejection curves for dataset eucalyptus.

sification accuracy should improve on the remaining instances. The dependency between the threshold t and the classification accuracy is typically depicted in the form of so-called *accuracy-rejection* curves.

In our experiments, we tested two very simple uncertainty indexes (needless to say, various other indexes are conceivable) directly related to the two types if uncertainty reflected by FR3: γ_c is the degree of conflict between the top-class as suggested by FR3 (in terms of the score (7)) and the second-best class. Likewise, γ_i is the degree of ignorance between these two classes. Again, each dataset was randomly split, in proportion 2:1, for training and testing. This was repeated 100 times, and each instance (occurring in potentially many of the 100 test sets) was associated with its average γ-index.

The monotonicity expected of the dependence between rejection threshold t and classification accuracy is confirmed by the experimental results (see [HH08] for details). Using γ_c, an improvement is obtained for all datasets, and γ_i leads to an improvement in all but three cases. Typical accuracy-rejection curves are shown in Fig. 3–3 (the plateaus in these curves are caused by the absence of instances with corresponding γ-values). In summary, these experiments clearly show that both measures of uncertainty derived by FR3, conflict and ignorance, are reliable indicators of the uncertainty involved in a classification decision.
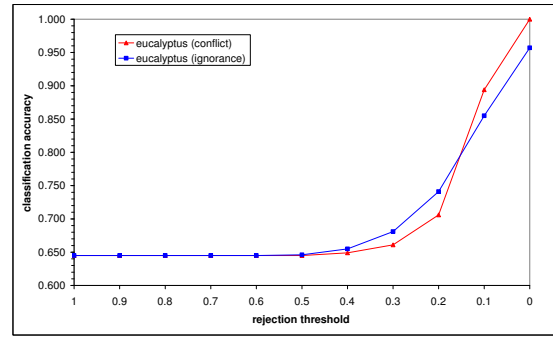
5 Concluding Remarks

In this paper, we have introduced a fuzzy rule-based classifier called Fuzzy Round Robin RIPPER (FR3). As opposed to conventional methods, FR3 carefully distinguishes between two sources of uncertainty in classification, namely conflict and ignorance, and, correspondingly, offers predictions of a more differentiated type: Against the background of the data seen so far, in conjunction with the underlying model assumptions, FR3 compares the potential decisions (class labels) in a pairwise manner and, for each pair, suggests to what extent one label is preferable to the other one, to what extent there is a conflict between these labels, and to what extent none of the two are supported. A prediction, or any other type of decision, can then be made on the basis of the fuzzy preference structure thus obtained.

Focusing on the core part of the method, namely the induction of the fuzzy preference structure, we have used relatively simple decision policies in this paper, both for standard classification (predicting a single class) and for classification with reject option. Nevertheless, developing suitable decision policies for different types of (generalized) classification problems is an important issue that we

plan to address in future work. An interesting idea, for example, is to employ techniques from belief function theory, which not only offers suitable means for representing ignorance, but also operators for combining different sources of information [QDM07].

A Java implementation of FR3, running under the open-source machine learning toolkit WEKA, can be downloaded at the following URL: `http://www.uni-marburg.de/fb12/kebi/research/software`.

Acknowledgments

References

[AN07]   A. Asuncion and D.J. Newman. UCI machine learning repository. http://www.ics.uci.edu/~mlearn/MLRepository.html, 2007.

[Bar07]   D. Barker. Dataset Pasture production. http://weka.sourceforge.net/wiki/index.php/Datasets, 2007. Obtained on 20th of October 2007.

[Bul07]   Bruce Bulloch. Dataset: Eucalyptus soil conservation. http://weka.sourceforge.net/wiki/index.php/Datasets, 2007. Obtained on 20th of October 2007.

[Coh95]   William W. Cohen. Fast effective rule induction. In Armand Prieditis and Stuart Russell, editors, *Proceedings of the 12th International Conference on Machine Learning*, pages 115–123, Tahoe City, CA, July 9–12, 1995. Morgan Kaufmann.

[Für02]   Johannes Fürnkranz. Round robin classification. *The Journal of Machine Learning Research*, 2:721–747, 2002.

[Für03]   Johannes Fürnkranz. Round robin ensembles. *Intell. Data Anal.*, 7(5):385–403, 2003.

[FR94]   J. Fodor and M. Roubens. *Fuzzy preference modelling and multicriteria decision support.* Kluwer Academic Publishers, 1994.

[FW94]   Johannes Fürnkranz and Gerhard Widmer. Incremental reduced error pruning. In *International Conference on Machine Learning*, pages 70–77, 1994.

[Har07]   Winna Harvey. Dataset: Squash harvest stored / unstored. http://weka.sourceforge.net/wiki/index.php/Datasets, 2007. Obtained on 20th of October 2007.

[HB08]   Eyke Hüllermeier and Klaus Brinker. Learning valued preference structures for solving classification problems. *Fuzzy Sets and Systems*, 2008. To appear.

[HH08]   J. Hühn and E. Hüllermeier. FR3: A fuzzy rule learner for inducing reliable classifiers. Technical report, Marburg University, Department of Mathematics and Computer Science, 2008.

[MV07]   Mike Meyer and Pantelis Vlachos. Statlib. http://lib.stat.cmu.edu/, 2007.

[QDM07]   B. Quost, T. Denoeux, and M.H. Masson. Pairwise classifier combination using belief functions. *Pattern Recognition Letters*, 28(5):644–653, 2007.

[WF05]   Ian H. Witten and Eibe Frank. *Data Mining: Practical machine learning tools and techniques.* Morgan Kaufmann, San Francisco, 2nd edition, 2005.