A Study of Heuristic Evaluation Measures in Fuzzy Rule Induction

Ashraf A. Afifi

A Study of Heuristic Evaluation Measures in Fuzzy Rule Induction

Ashraf A. Afifi

Department of Engineering, Design and Mathematics, Faculty of Environment and Technology, University of the West of England, UK
Industrial Engineering Department, Faculty of Engineering, Zagazig University, Egypt
Ashraf.Afifi@uwe.ac.uk

Abstract. The rule induction process could be conceived as a search process, and hence an evaluation metric is needed to estimate the quality of rules found in the search space and to direct the search towards the best rule. The evaluation measure is the most influential inductive bias in rule learning. It is therefore important to investigate its influence on the induction process and to compare the behaviour of different evaluation measures. Many different evaluation measures have been used to score crisp rules. For some of these measures, fuzzy variations have been designed and used to score fuzzy rules. This paper examines the most popular crisp evaluation measures and demonstrates how they can be adapted into the fuzzy domain. The paper also studies the performance of these measures on a large number of data sets when used in a recently developed fuzzy rule induction algorithm. Results show that there are no universally applicable evaluation measures and the choice of the best measure depends on the type of the data set and the learning problem.

Keywords: fuzzy rule induction, heuristic evaluation measures, fuzzy sets.

1 Introduction

Rule induction has proven to be a valuable tool for description, classification and generalisation of data. A variety of methods exist for rule learning using crisp sets [1-4]. Fuzzy sets are a generalisation of crisp sets providing increased expressive power and comprehensibility. A number of fuzzy rule induction algorithms have been developed as an extension of crisp rule learners [5, 6]. Recently, Afify [7] introduced a novel fuzzy rule induction method called FuzzyRULES (for Fuzzy RULe Extraction System). FuzzyRULES is based on the learning strategy used in the RULES-6 [8] crisp rule induction algorithm, previously developed by the author. FuzzyRULES preserves all the advantages of RULES-6 such as good comprehensibility and high classification accuracy. Moreover, it incorporates approximate reasoning offered by fuzzy representation, which allows better dealing with inconsistent, inexact, subjective, or noisy data.

Rule induction algorithms typically use an evaluation measure to score the performance of rules found during the learning process, and to select the best rules for further exploration. Therefore, the evaluation measure as a search heuristic is very important

as it determines to a large extent the performance of the learning algorithm. The selection of the most appropriate evaluation measure for any particular learning algorithm is a critical and difficult task, which is usually carried out empirically. This paper examines the role of several well-known crisp evaluation measures and how they address the specific requirements of the rule forming process. The paper also studies the performance of the fuzzy variants of these measures when applied to fuzzy rule induction algorithms. The aim is to identify the most appropriate measures to be applied in fuzzy rule induction algorithms, especially in FuzzyRULES.

The layout of the paper is as follows. Section 2 reviews the FuzzyRULES algorithm. Section 3 introduces the evaluation measures, and demonstrates the adaptation of these measures to the fuzzy set domain. Section 4 gives an empirical evaluation of the different evaluation measures when used in FuzzyRULES. Section 5 concludes the paper and provides suggestions for future work.

2 FuzzyRULES

The recently developed FuzzyRULES fuzzy rule induction algorithm will be used in this study to assess the performance of several heuristic evaluation measures. FuzzyRULES is based on the RULES-6 crisp rule induction algorithm. RULES-6 broadly follows the approach of AQ-like [2] learning algorithms. It employs heuristic search techniques with different learning biases and rule-space pruning strategies that significantly reduce the proportion of the search space examined during the learning process, resulting in substantial performance benefits.

FuzzyRULES follows the so-called *separate-and-conquer* or *covering* strategy of rule induction algorithms. It creates the rule set one rule at a time. Each rule explains (covers) a part of the training instances. After a rule is generated, the instances covered by it are removed (separated) from the training data set before subsequent rules are learned. The remaining instances are conquered using the same induction procedure until all the instances are covered by at least one rule in the rule set. Figure 1 provides a simplified description of the FuzzyRULES algorithm. A detailed description of the algorithms can be found in [7].

3 Evaluation Measures

A typical objective of a rule induction algorithm is to find rules that optimise a rule evaluation measure that takes both training accuracy and rule coverage into account so that the rules learned are both accurate and general. Many different measures for evaluating and assigning a score to crisp rules have been proposed in the literature [9]. For some of these measures, fuzzy variations have been designed and used to score fuzzy rules [10]. In this section, the most commonly used rule evaluation measures are discussed and the adaptation of these measures into the fuzzy domain is demonstrated.

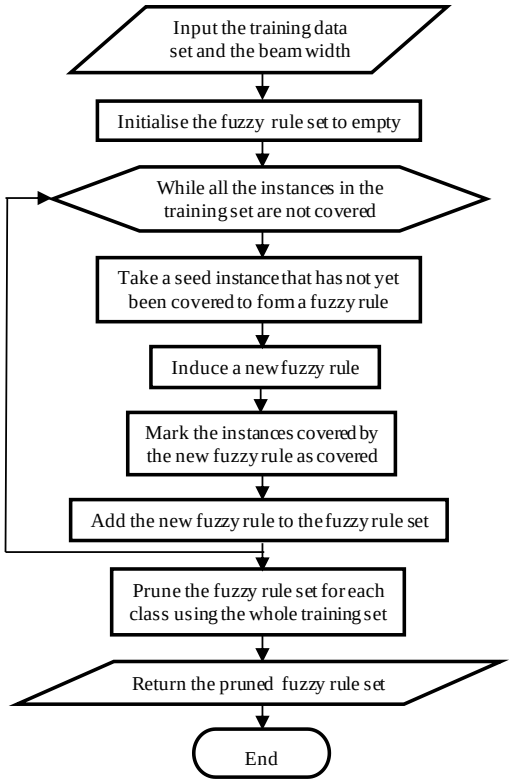

Fig. 1. Overview of the FuzzyRULES algorithm.

To derive fuzzy evaluation measures, the cardinality and α-cut operators [11] will be used to describe the instances matched by the rules. Let *A* => *C* denote a candidate rule *R*, where *A* is the rule antecedent (a conjunction of conditions) and *C* is the rule consequent (the value predicted for the goal attribute). The set of instances in the training set *T* that are covered by a rule *R* can be defined using the cardinality measure $M(A)$, also referred to as the sigma count, as follows:

$$M(A) = \sum_{I \in T} \mu_R(I) \tag{1}$$

where $\mu_R(I)$ represents the degree of match of a particular instance *I* with the antecedent of a rule *R*. For each instance, this degree is computed by a fuzzy AND of the degree of matching between each attribute value in the instance and the corresponding fuzzy condition in the rule. The conventional definition of the fuzzy AND has been used as the minimum operator.

In practice, many of the instances can match a rule antecedent to a small degree, and the summation of all these small degrees of membership can give the false impression that the rule is good, which can undermine the reliability of the evaluation measure. To prevent such instances from being covered, a user-defined α-cut threshold can be applied to the instance memberships. The membership of an instance to a rule, $\mu_R(I)$, is

defined to be zero when it lies below the α-cut threshold value. When the value of $\mu_R(I)$ is greater than or equal to the specified value of α, rule R is said to α-cover instance I. Using the α-cut operator, the set of instances in the training set T that are α-covered by a rule R can be expressed by redefining Equation (1) as follows:

$$M(A) = \sum_{I \in R_\alpha(T)} \mu_R(I) \tag{2}$$

where $R_\alpha(T) = \{I \in T | \mu_R(I) \geq \alpha\}$ contains all instances in the training set T that have a membership grade in R greater than or equal to the specified value of α.

The evaluation measures include different parameters that assess the coverage of the rules. For a particular rule R, the parameters P and N denote the total number of positive instances (instances belonging to the target class) and negative instances (instances not belonging to the target class) in the training set, p(n) the number of positive (negative) instances covered by rule R, and p' (n') the number of positive (negative) instances covered by rule R', the predecessor (parent) of rule R.

3.1 Purity

This measure is utilised in GREEDY3 [12] and SWAP-1 [13] algorithms. It gives the ratio between the number of positive instances covered by a rule and the total number of covered instances. In the crisp case, the purity measure of a rule R is given by:

$$P(R) = \frac{p}{p+n} \tag{3}$$

The purity measure attains its optimal value when no negative instances are covered. Also, it does not aim to cover many positive instances. As a result, this metric tends to select very specific rules covering only a small number of instances. This is undesirable since rules covering few instances are unreliable, especially where there is noise in the data. The accuracy of these rules on the training data does not adequately reflect their true predictive accuracy on new test data.

The purity measure can be fuzzified using the cardinality and α -cut operators as follows:

$$P(R) = \frac{M(A \cap C)}{M(A)} = \frac{\sum_{I \in R_\alpha(P)} \mu_R(I)}{\sum_{I \in R_\alpha(T)} \mu_R(I)} \tag{4}$$

where $M(A \cap C)$ is the fuzzification of the number of positive instances covered by a rule R (p), $M(A)$ is the fuzzification of the number of instances covered by the rule (n + p), and $R_\alpha(P) = \{I \in P | \mu_R(I) \geq \alpha\}$, $R_\alpha(T) = \{I \in T | \mu_R(I) \geq \alpha\}$ are the sets of positive instances and total instances that are α-covered by rule R, respectively.

3.2 Information Content

The information content function measures the amount of information contained in the classification of the covered instances. It is originally used in the PRISM inductive learner [14], and its crisp version is given by:

$$IC(R) = -\log_2\left(\frac{p}{p+n}\right) \tag{5}$$

The best rule is the one that minimises Equation (5). The information content function is basically equivalent to the purity measure and its main advantage is that using a logarithmic scale tends to assign higher penalties to rules with low coverage.

The fuzzy version of the information content function can be expressed as:

$$IC(R) = -\log_2\left(\frac{M(A\cap C)}{M(A)}\right) = -\log_2\left(\frac{\sum_{I\in R_\alpha(P)}\mu_R(I)}{\sum_{I\in R_\alpha(T)}\mu_R(I)}\right) \tag{6}$$

3.3 Entropy

The entropy measure is adopted in the original version of the CN2 algorithm [15]. It is the weighted average of the information content of the positive and negative classes. The entropy measure of a rule *R* is given by (the lower the entropy, the better the rule):

$$E(R) = -\left(\frac{p}{p+n}\log_2\frac{p}{p+n} + \frac{n}{p+n}\log_2\frac{n}{p+n}\right) \tag{7}$$

The entropy measure suffers from similar deficiencies as purity and information content measures. Also, it does not consider whether the majority of instances are positive or not. For example, a rule that covers 100 positive and 10 negative instances is considered of equal quality to another rule that covers 10 positive and 100 negative instances. To assign higher scores to rules with higher coverage, the following function is adopted [10].

$$E(R) = -\frac{n}{p}\left(\frac{p}{p+n}\log_2\frac{p}{p+n} + \frac{n}{p+n}\log_2\frac{n}{p+n}\right) \tag{8}$$

Equation (8) can be adapted to the domain of fuzzy sets as follows:

$$E(R) = -\frac{M(A\neg C)}{M(A\cap C)}\left(\frac{M(A\cap C)}{M(A)}\log_2\frac{M(A\cap C)}{M(A)} + \frac{M(A\neg C)}{M(A)}\log_2\frac{M(A\neg C)}{M(A)}\right) \tag{9}$$

$$= -\frac{\sum_{I\in R_\alpha(N)}\mu_R(I)}{\sum_{I\in R_\alpha(P)}\mu_R(I)}\left(\frac{\sum_{I\in R_\alpha(P)}\mu_R(I)}{\sum_{I\in R_\alpha(T)}\mu_R(I)}\log_2\frac{\sum_{I\in R_\alpha(P)}\mu_R(I)}{\sum_{I\in R_\alpha(T)}\mu_R(I)} + \frac{\sum_{I\in R_\alpha(N)}\mu_R(I)}{\sum_{I\in R_\alpha(T)}\mu_R(I)}\log_2\frac{\sum_{I\in R_\alpha(N)}\mu_R(I)}{\sum_{I\in R_\alpha(T)}\mu_R(I)}\right)$$

where $M(A\neg C)$ is the fuzzification of the number of negative instances covered by rule *R* (*n*), and $R_\alpha(N) = \{I \in N \,|\, \mu_R(I) \geq \alpha\}$ is the set of negative instances that are α-covered by rule *R*.

3.4 Information Gain

The information gain measure is originally used in the FOIL relational learner [16]. It computes the reduction of the information content in a rule when a new condition is added to it. More precisely, it calculates the difference in the information content of the current rule *R* and its parent R', multiplied by the number of covered positive instances as a bias for generality. In the crisp case, it is given by:

$$IG(R) = p \cdot \left(IC(R') - IC(R)\right) = p \cdot \left(-\log_2\left(\frac{p'}{p'+n'}\right) + \log_2\left(\frac{p}{p+n}\right)\right) \tag{10}$$

The goal is to maximise this measure.

The fuzzy information gain can be computed as follows:

$$IG(R) = p \cdot \left(\log_2\left(\frac{M(A' \cap C)}{M(A')}\right) - \log_2\left(\frac{M(A \cap C)}{M(A)}\right)\right) \tag{11}$$

$$= p \cdot \left(\log_2\left(\frac{\sum_{I \in R'_\alpha(P)} \mu_{R'}(I)}{\sum_{I \in R'_\alpha(T)} \mu_{R'}(I)}\right) - \log_2\left(\frac{\sum_{I \in R_\alpha(P)} \mu_R(I)}{\sum_{I \in R_\alpha(T)} \mu_R(I)}\right)\right)$$

where $M(A' \cap C)$ is the fuzzification of the number of positive instances covered by rule R' (p'), $M(A)$ is the fuzzification of the number of instances covered by the rule ($p' + n'$), and $R'_\alpha(P) = \{I \in P | \mu_{R'}(I) \geq \alpha\}$, $R'_\alpha(T) = \{I \in T | \mu_{R'}(I) \geq \alpha\}$ are the sets of positive instances and total instances that are α-covered by rule R', respectively.

3.5 Accuracy

This metric is employed in I-REP [17]. It is the proportion of positive instances that are covered (p) and negative instances that are not covered (N - n), in all instances (P + N). In the crisp case, the accuracy measure for a rule R is given by:

$$A(R) = \frac{p + (N - n)}{P + N} \tag{12}$$

The accuracy measure favours high coverage and scores rules in the range [-∞, ∞], with higher scores given to better evaluations. The problem of the accuracy measure is that this measure sometimes does not lead to a satisfactory behaviour. For example, it favours a rule that covers 2000 positive and 1000 negative instances over another rule that covers 1000 positive and only 1 negative instance.

A fuzzy variant of the accuracy measure can be defined as:

$$A(R) = \frac{M(A \cap C) + (N - M(A \neg C))}{P + N} = \frac{\sum_{I \in R_\alpha(P)} \mu_R(I) + (N - \sum_{I \in R_\alpha(N)} \mu_R(I))}{P + N} \tag{13}$$

3.6 Laplace

The Laplace estimate penalises rules with low coverage and it is used in CN2 [1] and several other algorithms due to its simplicity and efficiency. The Laplace estimate of a rule R is given by:

$$L(R) = \frac{p + 1}{p + n + k} \tag{14}$$

where k is the number of classes. Rule induction algorithms learn multi-class concepts by learning one class at a time. Thus, k is always 2.

The Laplace estimate has the desirable property of taking into account both accuracy and coverage when estimating rule accuracy. However, it has a problem when learning rules with less than 50% training accuracy. The Laplace estimate does not satisfy the requirement that the rule quality value should rise with increased coverage. Another problem is that the Laplace accuracy estimate is often unrealistic, especially in multi-class decision problems [18]. This occurred because of the assumption that underlies Laplace accuracy estimate, namely, that the *a priori* distribution is uniform.

A fuzzy variant of the Laplace estimate can be given by:

$$L(R) = \frac{M(A \cap C) + 1}{M(A) + 2} = \frac{\sum_{I \in R_\alpha(P)} \mu_R(I) + 1}{\sum_{I \in R_\alpha(T)} \mu_R(I) + 2} \tag{15}$$

3.7 *m*-estimate

The *m-estimate* [18] is a more general version of the Laplace measure and it is defined as follows:

$$M(R) = \frac{p + m * P/(P+N)}{p + n + m} \tag{16}$$

where $P/(P+N)$ is the *a priori* probability of the target class and m is a domain dependent parameter. The value of m is related to the amount of noise in the domain. m can be small if little noise is expected and should increase if the amount of noise is substantial. m is usually set to k, where k is the number of classes.

The *m*-estimate is a good choice because it is based on strong theoretical foundations and it meets the requirements of a good evaluation function. The Laplace measure can be obtained from the *m*-estimate when m is set to 2 and the *a priori* probability is assumed to be uniform. It should be noted that the *m*-estimate generalises the Laplace measure so that rules that cover no instances will be evaluated with the *a priori* probability instead of the value 1/2, which is more flexible and convenient.

The fuzzy variant of the *m*-estimate can be defined as follows:

$$M(R) = \frac{M(A \cap C) + k * P/(P+N)}{M(A) + k} = \frac{\sum_{I \in R_\alpha(P)} \mu_R(I) + k * P/(P+N)}{\sum_{I \in R_\alpha(T)} \mu_R(I) + k} \tag{17}$$

3.8 *H* Measure

This measure is applied in RULES-5 [3]. It is the product of the accuracy (first part) and generality (second part) of a rule.

$$H(R) = \sqrt{\frac{p+n}{P+N}} \left(2 - 2\sqrt{\frac{p}{p+n} \cdot \frac{P}{P+N}} - 2\sqrt{(1 - \frac{p}{p+n})(1 - \frac{P}{P+N})} \right) \tag{18}$$

The *H* measure can be fuzzified as follows:

$$H(R) = \sqrt{\frac{M(A)}{P+N}}\left(2 - 2\sqrt{\frac{M(A\cap C)}{M(A)}\cdot\frac{P}{P+N}} - 2\sqrt{(1-\frac{M(A\cap C)}{M(A)})(1-\frac{P}{P+N})}\right) \tag{19}$$

$$= \sqrt{\frac{\sum_{I\in R_\alpha(T)}\mu_R(I)}{P+N}}\left(2 - 2\sqrt{\frac{\sum_{I\in R_\alpha(P)}\mu_R(I)}{\sum_{I\in R_\alpha(T)}\mu_R(I)}\cdot\frac{P}{P+N}} - 2\sqrt{(1-\frac{\sum_{I\in R_\alpha(P)}\mu_R(I)}{\sum_{I\in R_\alpha(T)}\mu_R(I)})(1-\frac{P}{P+N})}\right)$$

3.9 AQ18 Measure

This measure was introduced in AQ18 [2] and it aims to combine completeness, consistency and consistency gain. This is done by allowing changes in the relative importance of completeness (first part) or consistency gain (second part) and also by changing the value of the parameter $w \in [0, 1]$.

$$Q(R) = \left(\frac{p}{P}\right)^w \cdot \left(\left(\frac{p}{p+n} - \frac{P}{P+N}\right)\cdot\frac{P+N}{N}\right)^{(1-w)} \tag{20}$$

The fuzzy version of the AQ18 measure can be expressed as:

$$Q(R) = \left(\frac{M(A\cap C)}{P}\right)^w \cdot \left(\left(\frac{M(A\cap C)}{M(A)} - \frac{P}{P+N}\right)\cdot\frac{P+N}{N}\right)^{(1-w)} \tag{21}$$

$$= \left(\frac{\sum_{I\in R_\alpha(P)}\mu_R(I)}{P}\right)^w \cdot \left(\left(\frac{\sum_{I\in R_\alpha(P)}\mu_R(I)}{\sum_{I\in R_\alpha(T)}\mu_R(I)} - \frac{P}{P+N}\right)\cdot\frac{P+N}{N}\right)^{(1-w)}$$

The aforementioned evaluation measures have been developed alongside specific algorithms and their performances have not been compared to other metrics when applied to the same rule induction algorithm. As part of this work, the performance of these measures when used in the FuzzyRULES algorithm is evaluated experimentally in the following section.

4 Experimental Evaluation

This section presents an empirical evaluation of the nine fuzzy heuristic evaluation measures discussed in the previous section. Experimental tests were carried out using the FuzzyRULES algorithm, without applying any pruning procedures because such post processing could mask the real effect of using one or another heuristic. Three criteria were used to evaluate the results, namely, classification accuracy, rule set complexity, and the number of rules examined to obtain the rule set. The complexity of a rule set is measured by the total number of conditions in that rule set. Tests were performed on 30 data sets from the UCI repository of machine learning databases [19]. The selected data sets either have only continuous attributes or a mixture of nominal and continuous attributes. The hold-out approach was used to partition the data into training and test data [20]. FuzzyRULES was run with the default settings, and the default value of 0.5 was used for the parameter *w* of the AQ18 measure.

Table 1 shows the classification accuracies obtained with each of the nine evaluation measures. Bold numbers indicate the best performance for a specific data set among the tested measures. In the last row, the total performance over all data sets is shown. A number of results are notable. First, the accuracy obtained with the *m*-estimate measure over all the data sets was in total higher than that produced with all the other measures. Also, the *m*-estimate measure produced the best classification results for 14 out of the 30 data sets. The Laplace and information content measures had a very similar total accuracy, but the Laplace measure had the best results overall for a higher number of data sets. The entropy and purity measures yielded the next best total accuracy, followed by the *H* and AQ18 measures. The accuracy and information gain measures obtained the lowest total accuracy. Second, every evaluation measure obtained the best results overall for some data sets, and the best performing measures had a notable worse performance for some other data sets. This suggests that there will not be a universal solution to all learning problems. The best result is obtained by choosing the evaluation measure that best fits the requirements of a particular learning problem.

Table 1. Classification accuracies for each evaluation measure when used in FuzzyRULES.

Data Set Name	Purity	Entropy	*Information Content*	*Information Gain*	Accuracy	Laplace	*m*-estimate	*H* Measure	*AQ18*
Abalone	23.0	24.2	23.0	15.8	21.6	22.9	23.9	**25.0**	18.8
Adult	**82.0**	79.4	81.9	79.8	77.1	81.9	81.8	79.4	80.1
Anneal	99.0	87.3	**99.3**	82.3	84.7	97.7	97.7	96.3	84.7
Australian	80.4	**87.0**	83.0	86.5	86.5	85.7	85.2	80.3	86.5
Auto	**72.5**	65.2	68.1	62.5	59.4	59.4	65.2	62.9	44.9
Balance-scale	64.6	64.6	64.6	56.9	66.0	64.6	**69.6**	65.0	66.0
Breast	**95.3**	94.8	94.0	92.7	92.7	92.3	94.3	91.9	92.7
Cleve	75.2	74.3	75.2	74.3	79.2	**82.2**	**82.2**	77.3	72.3
Crx	74.0	81.0	75.0	**83.0**	**83.0**	74.5	81.0	80.0	**83.0**
Diabetes	64.5	75.0	64.5	72.3	69.5	64.5	**75.2**	68.2	74.6
German	70.3	71.2	72.1	69.4	66.0	70.0	**75.6**	73.1	67.9
German-organisation	68.5	71.8	66.7	70.9	66.0	69.4	**76.4**	67.0	71.5
Glass	52.8	51.4	58.3	49.4	58.3	50.0	**64.7**	63.9	58.3
Glass2	78.2	**83.6**	76.4	76.4	76.4	**83.6**	**83.6**	72.7	81.5
Heart-disease	73.3	67.8	76.7	73.3	72.2	**80.0**	78.9	73.3	75.6
Heart-Hungarian	73.5	76.5	**98.5**	78.6	74.5	76.5	78.6	72.4	**75.5**
Hepatitis	80.8	82.7	78.8	73.1	**88.5**	86.5	84.8	82.7	**88.5**
Horse-colic	70.6	82.4	70.6	65.0	79.4	76.5	**83.5**	76.2	80.9
Hypothyroid	96.0	98.1	96.0	**98.8**	97.0	98.4	97.9	94.8	98.7
Ionosphere	93.2	88.5	92.3	87.8	95.3	92.3	92.3	88.9	**95.7**
Iris	**96.0**	94.0	**96.0**	94.0	94.0	**96.0**	**96.0**	96.0	**96.0**
Letter	80.5	76.4	82.2	53.2	71.1	**82.1**	66.1	60.7	58.3
Lymphography	74.0	**84.0**	78.0	56.0	68.0	78.0	76.0	78.0	64.0
Satimage	81.2	80.6	83.2	60.4	76.3	**86.0**	84.0	81.8	67.1
Segment	80.9	82.1	87.7	77.4	80.8	84.8	**89.5**	88.4	79.0
Shuttle	97.6	95.4	95.9	94.8	94.8	**98.4**	**98.4**	95.9	94.8
Sick-euthyroid	92.0	90.4	91.8	90.9	9.6	91.9	**96.4**	91.6	90.4
Sonar	70.0	70.0	**80.0**	74.3	77.1	80.0	**80.0**	68.6	61.4
Tokyo	91.5	91.3	92.3	91.9	87.3	**92.5**	92.1	91.4	90.8
Vehicle	63.1	48.3	63.5	51.7	61.3	64.5	**66.0**	61.7	53.2
Total	2314.4	2319.2	2365.6	2193.2	2213.8	2363.0	**2416.8**	2305.2	2252.8

Table 2 shows the total number of conditions generated with each of the evaluation measures. Bold numbers indicate the smallest number of conditions per rule set for a particular data set. The information gain measure produced significantly fewer conditions in total than the other measures. In addition, it obtained the smallest rule sets for 29 out of the 30 data sets. However, the over-general rule sets created resulted in a deteriorated classification accuracy for most data sets. The AQ18 measure also found simple rule sets but again had poor accuracy performance. The accuracy, *m*-estimate, *H*, entropy and Laplace measures created general rules without sacrificing accuracy for most data sets. These measures prefer rules covering more instances, which often helps the learning algorithm to prevent overfitting. The purity and information content measures generated the most complex rule sets, which might be an indication of overfitting. This is most likely due to that these measures prefer more consistent rules and do not favour rules with high coverage.

Table 3 presents the size of the search space examined for each evaluation measure. The bold numbers indicate the smallest search per data set. The number of rules explored during the search process is often directly linked to the rule set complexity. Therefore, the fewer and more general rules created by the information gain measure

Table 2. Total number of conditions for each evaluation measure when used in FuzzyRULES.

Data Set Name	Purity	Entropy	*Information Content*	*Information Gain*	Accuracy	Laplace	*m*-estimate	*H* Measure	AQ18
Abalone	105	56	103	**12**	325	108	53	36	15
Adult	814	137	1134	8	**7**	286	128	179	30
Anneal	60	24	47	**6**	33	49	39	36	63
Australian	222	6	207	**2**	**2**	126	118	52	**2**
Auto	69	59	62	**7**	32	62	48	67	39
Balance-scale	33	18	33	**2**	11	33	29	30	14
Breast	121	9	92	**3**	12	30	27	22	12
Cleve	103	30	78	**5**	10	52	52	47	17
Crx	173	17	159	**5**	13	141	151	50	13
Diabetes	50	34	48	**6**	9	39	32	26	9
German	402	125	352	**7**	73	195	173	341	41
German-organisation	536	338	471	**4**	175	222	219	312	45
Glass	142	79	124	**7**	34	104	60	96	14
Glass2	59	11	48	**4**	6	20	20	27	10
Heart-disease	136	16	63	**3**	6	106	60	47	15
Heart-Hungarian	82	4	72	**2**	12	42	38	34	16
Hepatitis	65	13	46	**4**	12	27	24	51	24
Horse-colic	135	16	104	**3**	8	121	107	134	13
Hypothyroid	89	20	61	**5**	25	52	49	7	18
Ionosphere	43	9	34	**2**	13	32	29	41	28
Iris	9	**3**	**5**	**3**	**3**	5	5	5	5
Letter	7863	3248	5434	**31**	1997	2573	1073	1307	517
Lymphography	76	17	61	**7**	9	54	29	63	17
Satimage	1844	395	1442	**24**	220	816	521	678	128
Segment	621	412	499	**13**	192	286	121	372	68
Shuttle	419	61	846	**19**	36	151	98	76	30
Sick-euthyroid	163	18	162	**9**	31	76	53	21	11
Sonar	49	15	34	**2**	16	32	31	26	17
Tokyo	89	13	74	**9**	15	82	55	77	28
Vehicle	406	204	402	**9**	169	360	158	150	61
Total	14978	5407	12297	**223**	3506	6282	3600	4410	1320

Table 3. Total number of rules for each evaluation measure when used in FuzzyRULES.

Data Set Name	Purity	Entropy	*Information Content*	*Information Gain*	Accuracy	Laplace	*m* -estimate	*H* Measure	AQ18
Abalone	1527	449	1500	**231**	3882	1560	936	752	424
Adult	38136	2920	30074	**643**	288	12639	14091	10543	1197
Anneal	2669	955	2089	**1320**	1898	2944	2310	2715	3659
Australian	4843	116	4393	**64**	53	3198	2938	2421	95
Auto	3195	2819	2812	**657**	1321	2802	2149	3507	1798
Balance-scale	179	92	179	**28**	59	177	161	157	67
Breast	1736	85	1216	**49**	127	400	364	239	135
Cleve	2683	730	2089	**174**	227	1539	1531	1529	352
Crx	5998	564	5619	**326**	333	4120	4496	1092	450
Diabetes	558	357	526	**181**	230	497	426	384	241
German	18651	6245	13952	**1156**	3836	8343	8475	12666	3210
German-organisation	28988	9890	25034	**934**	11778	14123	13134	17819	4644
Glass	2327	1434	2053	**198**	568	1807	977	1884	278
Glass2	753	134	636	**103**	**103**	291	294	449	143
Heart-disease	2390	395	1659	**101**	138	1559	1626	1595	329
Heart-Hungarian	1787	48	1671	**71**	181	1074	987	1084	406
Hepatitis	2186	406	1562	**207**	257	576	468	1817	653
Horse-colic	7338	540	6353	**216**	285	5841	4974	6370	560
Hypothyroid	2625	821	1948	964	1059	1533	1431	**399**	802
Ionosphere	2578	266	2029	**185**	687	1328	1207	1399	1098
Iris	50	**12**	30	**12**	**12**	23	23	23	18
Letter	326494	162820	241125	**2445**	114189	192745	77979	101101	19991
Lymphography	2678	531	2119	369	**325**	1947	687	2411	665
Satimage	163604	60538	128661	**1676**	14699	50811	37315	46844	9496
Segment	14062	10600	12389	**507**	5855	7077	4247	9308	2256
Shuttle	5548	755	11391	**165**	403	2412	1420	1033	270
Sick-euthyroid	4911	1666	4868	**963**	1655	3405	2849	1496	1046
Sonar	2906	1250	1292	**181**	735	1144	1038	979	849
Tokyo	7572	682	6633	**581**	629	5764	3231	5201	1387
Vehicle	14091	9633	14079	**705**	5555	13854	7082	8995	2209
Total	673063	277753	529981	**15412**	171367	345533	198846	246212	58728

made it to examine the least number of rules. It investigated 98% less rules than the purity method, which inspected the most rules. The amount of search required by the other evaluation measures is also related to the rule set complexity of these measures.

5 Conclusions and Future Work

This paper has examined the influence of different crisp evaluation metrics and analysed their role during the rule forming process. It has also proposed fuzzy variants of these measures and examined their performance when used in a recently developed fuzzy rule induction algorithm. The paper has demonstrated the strong effect of the evaluation measures on the performance of the learning algorithm. Evaluation measures that prefer more general over more specific rules helped the learning algorithm to determine the correct models for many data sets. However, over-generalisation may also deteriorate performance in problems that require a very specific concept description. The performance of additional evaluation measures, in particular those balancing the complexity and the accuracy of a particular induced model, could be studied on more data sets and other fuzzy rule induction algorithms.

Acknowledgements: The author wishes to thank the Department of Engineering, Design and Mathematics, University of the West of England for providing a good environment, facilities and financial means to complete this paper.

References

1. Clark, P., Boswell, R.: Rule induction with CN2: Some recent improvements. In: 5th European Conference on Artificial Intelligence, Porto, Portugal, pp. 151–163 (1991).
2. Michalski, R.S., Kaufman, K.A.: A measure of description quality for data mining and its implementation in the AQ18 learning system. In: ICSC Symposium on Advances in Intelligent Data Analysis, The Rochester Institute of Technology, USA, pp. 22–25, (1999).
3. Pham, D.T., Bigot, S., Dimov, S.S.: RULES-5: A rule induction algorithm for problems involving continuous attributes. Proceedings of the Institution of Mechanical Engineers, Part C: J. Mech. Eng. Science, **217**(12), 1273–1286 (2003).
4. Pham, D.T., Afify, A.A.: SRI: A scalable rule induction algorithm. Proceedings of the Institution of Mechanical Engineers, Part C: J. Mech. Eng. Science, **220**(5), 537-552 (2006).
5. Hühn, J., Hüllermeier, E.: FURIA: An algorithm for unordered fuzzy rule induction. Data Mining and Knowledge Discovery, **19**, 293–319 (2009).
6. Afify, A.A.: A novel algorithm for fuzzy rule induction in data mining. Proceedings of the Institution of Mechanical Engineers, Part C: J. Mech. Eng. Science, **228**(5), 877–895 (2014).
7. Afify, A.A.: A fuzzy rule induction algorithm for discovering classification rules. J. Intelligent and Fuzzy Systems, **30**(6), 3067-3085 (2016).
8. Pham, D.T., Afify, A.A.: RULES-6: A simple rule induction algorithm for handling large data sets. Proceedings of the Institution of Mechanical Engineers, Part C: J. Mech. Eng. Science, **219**(10), 1119–1137 (2005).
9. Fürnkranz, J., Flach, P.A.: An analysis of rule evaluation metrics. In: 20th International Conference on Machine Learning, Washington, DC, USA, pp. 202–209 (2003).
10. van Zyl, J., Cloete, I.: Heuristic functions for learning fuzzy conjunctive rules. In: IEEE International Conference on Systems, Man and Cybernetics, The Hague, The Netherlands, pp. 2332–2337 (2004).
11. Klir, G.J., Yuan, B.: Fuzzy Sets and Fuzzy Logic: Theory and Applications, Prentice-Hall, Upper Saddle River, New Jersey, USA (1995).
12. Pagallo, G., Haussler, D.: Boolean feature discovery in empirical learning. Machine Learning, **3**, 71–99 (1990).
13. Weiss, S., Indurkhya, N.: Reduced complexity rule induction. In: 12th International Joint Conference on Artificial Intelligence, Sydney, Australia, pp. 678–684 (1991).
14. Cendrowska, J.: PRISM: An algorithm for inducing modular rules. International J. Man-Machine Studies, **27**, 349–370 (1987).
15. Clark, P., Niblett, T.: The CN2 induction algorithm. Machine Learning, **3**, 261–284 (1989).
16. Quinlan, J.R.: Learning logical definitions from relations. Mach. Lear., **5**, 239–266 (1990).
17. Fürnkranz, J., Widmer, G.: Incremental reduced error pruning. In: 11th International Conference on Machine Learning, New Brunswick, NJ, USA, pp. 70–77 (1994).
18. Cestnik, B.: Estimating probabilities: A crucial task in machine learning. In: 3rd European Conference on Artificial Intelligence, Stockholm, Sweden, pp. 147–149 (1990).
19. Blake, C.L.,Merz, C.J.: UCI Repository of Machine Learning Databases. http://www.ics.uci.edu/~mlearn/MLRepository.html, last accessed 2017/6/1.
20. Efron B., Tibshirani R.: An Introduction to the Bootstrap. Chapman & Hall, USA (1993).