Evolving Systems (2016) 7:255–265
DOI 10.1007/s12530-015-9142-4

ORIGINAL PAPER

# On fuzzy feature selection in designing fuzzy classifiers for high-dimensional data

**Eghbal G. Mansoori[1] · Khadijeh S. Shafiee[1]**

Received: 28 August 2015 / Accepted: 19 November 2015 / Published online: 30 November 2015


**Abstract** Generating fuzzy rules for high-dimensional data has been a serious challenge in designing fuzzy rule-based classification systems. For data sets with low dimensions, there are some efficient methods to generate a compact set of short fuzzy rules. However, when the dimensions go up, the number of rules increases exponentially. One solution for lowering the dimensions is feature selection which selects a subset of more effective features. In this regard, a fuzzy feature selection approach is proposed in this paper which tries to choose more relevant features; those which can distinguish the distinct classes well. Our method employs the training patterns in the subspace of some predefined fuzzy sets on each feature and applies their compatibility degrees to evaluate that feature. Since each feature is evaluated individually, this method can be applied efficiently on high-dimensional data. Using the selected features to generate rules in fuzzy rule-based classifiers, this paper also presents a novel criterion to assess each generated rule. This criterion measures the capability of each fuzzy rule in discriminating the positive and negative patterns. To illustrate the scalability of our fuzzy feature selection method beside to the efficiency of generated fuzzy rules, they are applied on some benchmark data sets and the results are compared to some other methods in the literature. The experimental results justify the feasibility of our approach to work with high-dimensional data and its acceptable performance in terms of designing CPU time and classification accuracy.

**Keywords** Fuzzy rule-based classifier · Fuzzy rule · Feature selection · Fuzzy feature selection

## 1 Introduction

Fuzzy models are developed by fuzzy rule-based classification systems, where the output of systems is crisp and discrete. The possibility to work with imprecise data and missing values, and also, human understandable form of the acquired knowledge, are the main advantages of fuzzy models (Mansoori et al. 2008; Marin-Blazquez and Shen 2002). Basically, the design of a fuzzy rule-based classifier tries to find a compact set of fuzzy if–then rules to be able to model the input–output behavior of the system. Many approaches for generating fuzzy classification rules from data have been proposed in the literature. These methods include heuristic approaches (Mansoori et al. 2007; Ishibuchi and Yamamoto 2004; Ishibuchi and Nakashima 2001), neuro-fuzzy techniques (Nauck and Kruse 1997; Almaksour and Anquetil 2011), association rule discovery (Alcala-Fdez et al. 2011a), genetic algorithm (Mansoori et al. 2008), and based on evolving systems (Iglesias et al. 2010; Lughofer and Buchtala 2013; Angelov et al. 2008).

In high-dimensional problems, the rule base of a fuzzy classification system would have too many rules (Rehm et al. 2007). So, reducing the search space of fuzzy rules in designing phase of classifiers is an important concern. In several researches, many methods have been suggested for solving this problem so far. For example, rule reduction methods using neural networks (Halgamuge and Glesner 1994), clustering techniques (Chiu 1994) and similarity measures (Setnes et al. 1998) have been recommended. Also, there have been GA-based methods for selecting a set of cooperative rules among the set of candidate rules

✉ Eghbal G. Mansoori
mansoori@shirazu.ac.ir

1 School of Electrical and Computer Engineering, Shiraz University, Shiraz, Iran

(Cordon et al. 1999; Roubos and Setnes 2000). Feature weighting is another technique for decreasing softly and smoothly the influence of features in the rules (Lughofer 2011). However in high-dimensional problems, obtaining a small and efficient rule base is difficult and the interpretability of the system could not be guaranteed (Alcala-Fdez et al. 2011a; Cassillas et al. 2001).

Another suggested approach for reducing the search space is to generate some fuzzy rules with restricted antecedent conditions (Ishibuchi and Murata 1997). In this regard, some effective features from a high-dimensional problem should be selected. Feature selection or data dimensionality reduction refers to the process of identifying a few, yet more important, variables or features which help in predicting the outcomes. There are many potential benefits for feature selection. These include facilitating data visualization and data understanding, reducing the measurement and storage requirements, decreasing the training and utilization times and avoiding the curse of dimensionality to improve the prediction performance.

In general, the feature selection methods can be grouped in three categories: filters, wrappers and embedded models. Filters are used to score all features via a preprocessing stage and then select the best ones. In wrappers, some feature sets are selected and then evaluated via the designed classifiers. The embedded methods, however, are specific to the selected learning machines (Guyon and Elisseeff 2003; Tuv et al. 2009) and the process of feature selection is done in their training step. Some of the common feature selection approaches include: Fischer criterion (Fisher 1936), fuzzy entropy (Lee et al. 2001; Shie and Chen 2007) and similarity measure (Luukka 2011), mRMR method (Peng et al. 2005), and mutual information (MI) (Estevez et al. 2009). However, computing Shannon's MI between high-dimensional vectors is impractical because the number of samples and so the required CPU time is high (Estevez et al. 2009).

Feature selection methods can also be viewed from another perspective. Traditional algorithms select the features for all classes in common while class-specific feature selection approaches try to find a subset of features for each class separately. Using class-specific feature selection methods, a better discrimination of classes have been resulted in most of cases (Pineda-Bautista et al. 2011). Also, recently some feature selection methods have been proposed which combine fuzzy and other approaches. Neuro-fuzzy solutions (Chakraborty and Pal 2004) or genetic feature selection methods (Yang and Honavar 1998; Casillas et al. 2001) constitute most of the researches in this field. However, computational complexity is their major difficulty.

Impressing the second preference of fuzzy models, we have proposed a fuzzy feature selection algorithm in this paper. Its aim is to choose the more relevant features; those which can distinguish the distinct classes well. Our method is a class-specific approach which tries to find a subset of features for each class separately. It combines the interclass distance concept [as in Fisher (1936)] with the compatibility degree of data in some predefined fuzzy sets on each feature to evaluate that feature. Since our method processes each feature individually, it can be applied efficiently on high-dimensional data.

Our approach also selects some suitable fuzzy sets for each dimension in order to have a good (small and so more interpretable) set of rules. Moreover, a new criterion for evaluating the capability of each candidate rule in discriminating the positive and negative patterns is also introduced. It leads to select more powerful rules which result in a more efficient rule base.

The rest of this paper is organized as follows. In Sect. 2, the general design of fuzzy rule-based classification systems is explained. Our fuzzy method for feature selection is described in Sect. 3. In Sect. 4, we explain our method for designing fuzzy classifiers. The experimental results are presented in Sect. 5. Section 6 concludes the paper.

## 2 General design of fuzzy rule-based classification systems

Consider a classification problem with a data set of $m$ patterns, $DS = \{(X_p;y_p),\ p = 1..m\}$. For $p$th pattern, the input vector of variables, $X_p$, is $n$-dimensional. That is, $X_p = [x_{p1}, \ldots, x_{pn}]$ with feature labels $\{f_i,\ i = 1, \ldots, n\}$. The output variable, $y_p$, is a class label in $M$ classes such that $y_p \in \{c_1, \ldots, c_M\}$. We assume that each input variable, $x_{pi}$, is rescaled to unit interval [0,1] using a linear transformation that preserves the distribution of the data set.

In this paper, the classical single model architecture of fuzzy classifiers is utilized to handle the multiclass classification problems. The benefits of this model are simplicity, transparency and more interpretability of the designed classifiers (Lughofer and Buchtala 2013). In this model, the general form of fuzzy if–then rules is:

$$\begin{aligned}&\text{Rule } R_j:\ \text{if } x_1 \text{ is } A_{j1} \text{ and } \ldots\\ &\quad \text{and } x_n \text{ is } A_{jn} \text{ then class } C_j, \quad \text{for } j = 1, \ldots, N \end{aligned} \tag{1}$$

where $X = [x_1, \ldots, x_n]$ is an input vector, $A_{ji}$ $(i = 1, \ldots, n)$ indicates the fuzzy set on variable $x_i$ in the antecedent part of $R_j$, $C_j$ is the consequent class (that is, $C_j \in \{c_1, \ldots, c_M\}$), and $N$ is the number of fuzzy rules. Herein, the fuzzy rule $R_j$ is abbreviated as $A_j \Rightarrow$ class $C_j$ where $A_j = A_{j1} \times \cdots \times A_{jn}$. Generally, designing a fuzzy classifier can be described as generating a set of $N$ fuzzy rules in the form of (1).

The first step in generating fuzzy rules is partitioning the pattern space into fuzzy subspaces. If a subspace contains some patterns, a fuzzy rule will refer to it. Partitioning is usually done using $K$ suitable membership functions. The most common type of membership functions is triangular because they are simpler and easily understandable by humans. Moreover under some assumptions, the fuzzy partitions built out of the triangular membership functions lead to entropy equalization (Pedrycz 1994). Figure 1 shows these membership functions for four different values of $K$. Though up to five membership functions is common in generating fuzzy classification rules, the number of entities a human can reliably handle is seven to nine at most. So, this is often used as upper bound on the fuzzy sets in fuzzy modeling techniques (Gacto et al. 2011).

For the problem of generating fuzzy classification rules, some approaches have been suggested in Mansoori et al. (2007) and Ishibuchi and Yamamoto (2004). The approach in Ishibuchi and Yamamoto (2004) applies the fuzzy set *don't care* (with membership function $\mu_{don't\,care}(x_i) = 1, \quad \forall x_i \in [0,1]$) beside the 14 triangular fuzzy sets in Fig. 1. Using this *don't care* fuzzy set for a variable in the antecedent part of a rule will have that variable to be removed and so reduce the length of rule.

The consequent class $C_j$ of fuzzy rule $R_j$ in (1) is determined using the patterns in the corresponding fuzzy subspace. The compatibility grade of training pattern $X_p = [x_{p1}, \ldots, x_{pn}]$ is defined with the antecedent part $A_j = A_{j1} \times \cdots \times A_{jn}$ of rule $R_j$ as:

$$\mu_j(X_p) = \prod_{i=1}^{n} \mu_{ji}(x_{pi}) \tag{2}$$

where $\mu_{ji}(x_i)$ is the membership function of the antecedent fuzzy set $A_{ji}$ on variable $x_i$. One of the methods for selecting the consequent class of a rule is based on confidence [Ishibuchi and Yamamoto 2004] Bouchachia and Mittermeir 2006. The confidence of the fuzzy rule $A_j \Rightarrow \text{class}\, c$ is defined as:

$$Conf\,(A_j \Rightarrow \text{class } c) = \frac{\sum_{X_p \in \text{class } c} \mu_j(X_p)}{\sum_{p=1}^{m} \mu_j(X_p)}. \tag{3}$$

The consequent class $C_j$ of fuzzy rule $R_j$ can be obtained by identifying the class with the maximum confidence as:

$$C_j = \arg\max_{c}\{Conf\,(A_j \Rightarrow class\; c) | c \in \{c_1, \ldots, c_M\}\}. \tag{4}$$

In Ishibuchi and Yamamoto (2004), some heuristic measures for evaluating the candidate rules have been used. A basic criterion is the difference between the number of positives and negative samples. Its fuzzy version is specified as:

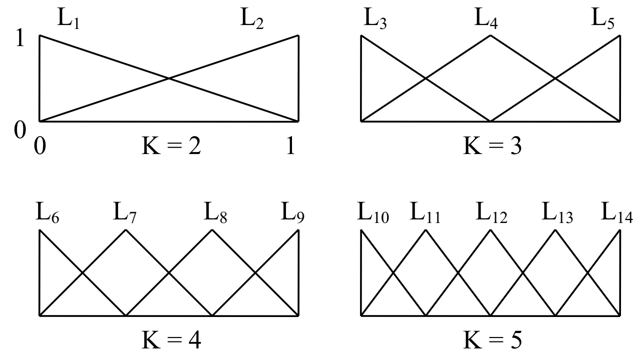


**Fig. 1** Fourteen fuzzy sets of each input variable

$$Eval(A_j \Rightarrow \text{class } C_j) = \sum_{X_p \in \text{ class } C_j} \mu_j(X_p) - \sum_{X_p \notin \text{ class } C_j} \mu_j(X_p). \tag{5}$$

Single winner (that is, winner-takes-all approach) is the most popular reasoning method in fuzzy rule-based classifiers (Ishibuchi et al. 1999) because of its simplicity and intuition for human users. Using this method, a new pattern $X_t = [x_{t1}, \ldots, x_{tn}]$ is classified according to the consequent class of the winner rule $R_w$. Indeed, the winner rule has the maximum compatibility grade with $X_t$ among the fired rules. This can be stated as:

$$\mu_w(X_t) = \max\{\mu_j(X_t), j = 1, \ldots, N\} \tag{6}$$

where $\mu_j(X_t)$ is the compatibility grade of rule $R_j$ with pattern $X_t$ in (2).

## 3 The proposed fuzzy feature selection method

The basis of our method is using the distribution of patterns in the fuzzy sets which are applied on each dimension (feature). This hopefully will obtain more relevant and interpretable features to be used in fuzzy rule-based classifiers. To avoid the curse of dimensionality problem, the features are selected for each class individually. Generally, more relevant features are those which can better discriminate the different classes. The basic idea is that a feature is relevant to a class if the number of patterns with true class labels (true positives) is more than the others (false positives). So, the more the difference of the summation of membership degree of positive and negative patterns in the fuzzy sets is, the more the feature is relevant to the positive class. The relevance degree of feature $f_i$ to class $c$ in $l$th fuzzy set is defined as:

$$\text{R}el\,ev\,(f_i, c, l) = \sum_{X_p \in \text{ class } c} \mu_{L_l}(x_{pi}) - \sum_{X_p \notin \text{ class } c} \mu_{L_l}(x_{pi}) \tag{7}$$

where $L_l$ is one of the fuzzy sets in Fig. 1. Thus, the effectiveness of this feature in class $c$ can be calculated by summing up the measures in (7) for all fuzzy sets in Fig. 1, as:

$$Effec\,(f_i, c) = \sum_{l=1}^{14} \{\mathrm{R}elev\,(f_i, c, l) | \mathrm{R}elev\,(f_i, c, l) > 0\}. \quad (8)$$

Clearly, only the fuzzy sets with positive relevance degree in (7) are contributed in computing the effectiveness measure of each feature. This feature selection approach has been summarized in the following algorithm. Its computational complexity is $O(nM)$ for $n$ features and $M$ classes in addition to ranking complexity of features and fuzzy sets, $O(n\ logn) + O(16\ n')$, which sums up to $O(n\ logn)$ since $n' \ll n$ and $14 \times \log 14 \approx 16$.

**Algorithm:** fuzzy feature selection

**Inputs:** $m$ data patterns of an $n$-dimensional $M$-class problem: $DS=\{(X_p,y_p) \mid X_p=[x_{p1}, \ldots, x_{pn}],$ $y_p \in \{c_1, ..., c_M\}, p=1..m\}$; feature labels: $\{f_1, \ldots, f_n\}$; number of desired features: $n'$; number of fuzzy sets on each feature: $K'$

**Outputs:** $n'$ selected best features for each class and their $K'$ more effective fuzzy sets

**for** each class $c_k$, $k$=1..$M$

**for** each feature $f_i$, $i$=1..$n$

$r_{i,l} = Relev\,(f_i, c_k, l),\ l = 1..14$

$e_i = Effec\,(f_i, c_k) = \sum_{l=1}^{14} \{r_{i,l} \mid r_{i,l} > 0\}$

**end for**

Rank, in descending order of $e_i$'s, the $n$ features: $f_{t_1} \rangle ... \rangle f_{t_i} \rangle ... \rangle f_{t_n},\ t_i \in \{1,...,n\}$

Select the top $n'$ features: $\{f_{t_1}, ..., f_{t_{n'}}\}$

Report $\{f_{t_1}, ..., f_{t_{n'}}\}$ as the best features for class $c_k$

**for** each selected feature $f_{t_j}$, $j = 1..n'$

Rank, in descending order of $r_{t_j,l}$'s, the fuzzy sets: $L_{t_j,s_1} \rangle ... \rangle L_{t_j,s_l} \rangle ... \rangle L_{t_j,s_{14}},\ s_l \in \{1,...,14\}$

Select the top $K'$ fuzzy sets (if any): $\{L_{t_j,s_1}, ..., L_{t_j,s_{K'}}\}$

Report $\{L_{t_j,s_1}, ..., L_{t_j,s_{K'}}\}$ as the best fuzzy sets for feature $f_{t_j}$ in class $c_k$

**end for**

**end for**

**stop**

Since the features are ranked according to their effectiveness values and then the most important ones are selected, this algorithm performs a single dimension-wise feature selection step in a kind of greedy-like manner. So, it can be trapped in local optima and therefore, only truly redundant features can be discarded. A greedy method finds the global optimal solution only when a feature is completely unimportant, but may get important when joined with another feature in two-dimensional space (Guyon and Elisseeff 2003).

## 4 Our method for designing fuzzy rule-based classifiers

To generate the fuzzy classification rules (as candidates which should be evaluated in next phase), the method in Ishibuchi and Yamamoto (2004) is used. This approach simultaneously considers all membership functions in Fig. 1 for each variable (feature). That is, one of the 14 fuzzy sets beside the *don't care* is used for each variable when generating a candidate rule. This can reduce the number of antecedent fuzzy sets of each rule. But instead of employing all 14 fuzzy sets for each variable in our approach, only $n'$ selected features and their $K'$ suitable fuzzy sets (which are identified by fuzzy feature selection algorithm) beside to *don't care* are used. So, the length of rules would be $n'$, at most. Moreover, since the features and their fuzzy sets are class-specific, the consequent of generated rules are predefined to that class. In other words, the combination of $K'$ fuzzy sets, identified for each of $n'$ features (of class $c$), will construct the antecedent part of the rules while their consequent part is set to class $c$. Thus, the

**Table 1** Data sets used in the experiments

| Data set | No. of features ($n$) | No. of patterns ($m$) | No. of classes ($M$) |
|---|---|---|---|
| Iris | 4 | 150 | 3 |
| Bupa | 6 | 345 | 2 |
| Ecoli | 7 | 336 | 8 |
| Pima | 8 | 768 | 2 |
| Yeast | 8 | 1484 | 10 |
| Cancer | 9 | 684 | 2 |
| Glass | 9 | 214 | 6 |
| Vowel | 10 | 990 | 11 |
| Wine | 13 | 178 | 3 |
| Image | 18 | 210 | 7 |
| Vehicle | 18 | 846 | 4 |
| Segment | 18 | 2310 | 7 |
| Ionosphere | 33 | 351 | 2 |
| Sonar | 60 | 208 | 2 |
| Coil2000 | 84 | 1220 | 2 |
| Musk | 166 | 476 | 2 |
| Fox | 230 | 476 | 2 |
| Tiger | 230 | 1220 | 2 |
| Secom | 590 | 1567 | 2 |
| Cnae9 | 856 | 1080 | 9 |

number of generated fuzzy rules for each class would be $K'^{n'}$ (at most), where $K' < 14$ and $n' \ll n$ for high-dimensional data. However, a fuzzy rule with a specific consequent class $c$ is generated only if the number of positive patterns (from class $c$) is more than negative patterns.

After generating the candidate rules, the next step is to construct the rule base among the candidates. Since the interpretability of rules is a major issue in fuzzy rule-based classifiers, the final rule base should be as compact as possible (Lughofer et al. 2011). For this purpose, the candidate rules should be evaluated and the best ones are selected. Several heuristic criteria have been suggested so far (Mansoori et al. 2007, 2008) and there is a good survey on some of these metrics in Ishibuchi and Yamamoto (2004).

By introducing covering subspace and decision subspace for each fuzzy rule in Mansoori et al. (2007), the authors proposed two thresholds for identifying these two subspaces. In this regard, the patterns having positive membership degree are considered to reside in covering subspace of a rule. On the other hand, those patterns with membership degrees greater than 0.5 are used to determine the decision subspace since they will certainly be classified by this rule. Using only the patterns in the decision subspace of a rule, we have proposed a modified version of criterion in (5) for candidate rule evaluation. This new measure can be formulated as:

**Table 2** The effectiveness measure of features in Iris data set

| Class | Features | | | |
|---|---|---|---|---|
| | Petal width | Petal length | Sepal width | Sepal length |
| Setosa | 133.0 | 126.2 | 65.6 | 36.1 |
| Versicolour | 63.7 | 44.9 | 14.5 | 0.6 |
| Virginica | 85.4 | 67.8 | 39.2 | 0.0 |

$$Eval'(R_j) = Eval'(A_j \Rightarrow \text{class } C_j) = \sum_{X_p \in \text{ class } C_j} \mu'_j(X_p) - \sum_{X_p \notin \text{ class } C_j} \mu'_j(X_p) \tag{9}$$

where

$$\mu'_j(X_p) = \prod_{i=1}^{n} \{\mu_{ji}(x_{pi}) | \mu_{ji}(x_{pi}) > 0.5\}. \tag{10}$$

To construct the final rule base, all candidate rules which are generated for each class in the first step are ranked and some best rules are chosen. For this purpose, a simple hill climbing method is used. In this regard, firstly the best rule for each class according to (9) is considered as rule base. Then, the next best and most cooperative rules for all classes are added to the rule base repeatedly in a greedy manner according to the classification accuracy of rule base on the training data. The accuracy of classifier using fuzzy rule base $RB$ on data set $DS$ is defined as:

$$Acc(DS, RB) = \frac{\sum_{k=1}^{M} m_k}{|DS|} \tag{11}$$

where $m_k$ is the number of patterns from class $c_k$ in $DS$ that are classified truly by using fuzzy rule base $RB$. This algorithm is explained here.

To obtain the complexity of algorithm, the required computations in each step is accounted. In step 1, the loop runs $M$ times while in each iteration, $K'^{n'}$ fuzzy rules are generated (for $n'$ selected features, $K'$ fuzzy sets and $M$ classes). So in step 1, the complexity is $O(MK'^{n'})$. This is also the complexity of step 2, since for each of $M$ classes, $K'^{n'}$ rules are evaluated. To classify $m$ patterns in step 3, $K'^{n'}$ rules are used so, this step needs $O(MK'^{n'})$. In steps 4–7, the computations are not noticeable, except in step 6 where requires the step 3 to be repeated some times. In overall, the complexity of algorithm is $O\Big((m + 2M)K'^{n'}\Big) \approx O(mK'^{n'})$ since in real-world data sets $m \gg M$. However, $K'$ and $n'$ are set to 4 in the experiments, so the complexity is $O(256m)$, in practice.

**Algorithm**: Designing a fuzzy rule-based classifier

**Inputs**: $m$ data patterns of an $n$-dimensional $M$-class problem: $DS=\{(X_p,y_p) \mid X_p=[x_{p1}, \ldots, x_{pn}],$
$y_p \in \{c_1, \ldots, c_M\}, p=1..m\}$; $n'$ selected features: $\{f_{t_j}, j=1..n'\}$, and their $K'$ fuzzy sets:
$\{L_{t_j,s_1}, \ldots, L_{t_j,s_{K'}}\}$; number of rules for each class: $Q$

**Outputs**: Final rule base: $FRB$

1) Generate $Q$ fuzzy rules for each class as candidates: $CRB$
   Let $CRB = \{\}$
   **for** each class $c_k$, $k$=1..$M$
      Generate all fuzzy rules for class $c_k$ using features $\{f_{t_j}, j=1..n'\}$ and sets $\{L_{t_j,s_1}, \ldots, L_{t_j,s_{K'}}\}$
      Rank, in descending order of their measures in (9), the generated rules
      Select the top $Q$ of the rules: $AFR$
      $CRB = CRB + AFR$
   **end for**
2) Initialize the rule base: $RB$
   Let $RB = \{\}$
   **for** each class $c_k$, $k$=1..$M$
      $R_{best} = \text{argmax } \{Eval'(R_j) \mid R_j \in CRB \text{ and } C_j = c_k\}$
      $RB = RB + \{R_{best}\}$
      $CRB = CRB \setminus \{R_{best}\}$
   **end for**
   Let $U = \{\}$
3) Classify data set $DS$ using rule base $RB$, then use (11) to compute overall accuracy, $acc$, and accuracy of each class, $acc_k$, $k \in \{1, \ldots, M\} \setminus U$
4) Find the class with the worst accuracy
   $c_{worst} = \text{argmin } \{acc_k, k \in \{1,\ldots,M\} \setminus U\}$
5) Enrich the rule base with the best rule from class $c_{worst}$
   $R'_{best} = \text{argmax } \{Eval'(R_j) \mid R_j \in CRB \text{ and } C_j = c_{worst}\}$
   $RB' = RB + \{R'_{best}\}$
   $CRB = CRB \setminus \{R'_{best}\}$
   Classify data set $DS$ using $RB'$, then use (11) to compute accuracy, $acc'$
   **if** $acc' > acc$
      Replace $RB$ with $RB'$
   **else**
      $CRB = CRB + \{R'_{best}\}$
      $U = U + \{c_{worst}\}$
   **end if**
6) Repeat steps 3-5 until $U$ is empty or no change in $RB$
7) $FRB = RB$
8) Return FRB

**Table 3** The best fuzzy sets of two best features for each class of Iris

| Setosa | | Versicolour | | Virginica | |
|---|---|---|---|---|---|
| Petal length | Petal width | Petal length | Petal width | Petal length | Petal width |
| $L_3$ | $L_3$ | $L_{12}$ | $L_{12}$ | $L_{13}$ | $L_5$ |
| $L_6$ | $L_6$ | $L_4$ | $L_4$ | $L_5$ | $L_{13}$ |
| $L_{10}$ | $L_{11}$ | $L_7$ | $L_7$ | $L_9$ | $L_9$ |
| $L_{11}$ | – | – | – | $L_{14}$ | $L_{14}$ |

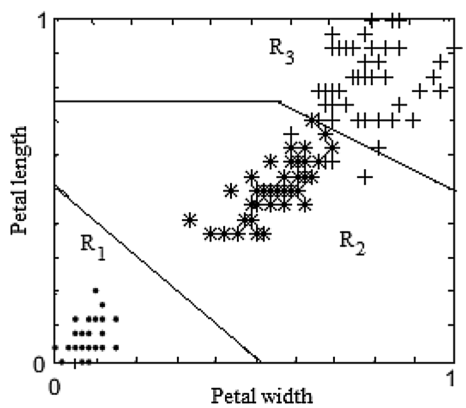


**Fig. 2** Decision area of three rules generated for Iris data set

## 5 Experimental results

In this section, the efficiency of the proposed methods is examined. The results are obtained by applying our algorithms on 14 data sets with low and moderate dimensions and 6 high-dimensional ones, all from UCI ML repository (Asuncion and Newman 2007). Table 1 summarizes the data used in the experiments, ranked in their number of features.

The Iris data set is used to illustrate the steps of our method. This data set consists of 150 samples with four dimensions and three classes. The effectiveness measure in (8) for its features is shown in Table 2. Table 3 illustrates at most four of the best fuzzy sets (in Fig. 1) for two best features of each class.

After applying our proposed method, the final rule base will contain three fuzzy rules. Figure 2 depicts the decision area of these rules.

$R_1$: If Petal width is $L_3$ and Petal length is $L_3$ then class is Setosa.
$R_2$: If Petal width is $L_4$ and Petal length is $L_4$ then class is Versicolour.
$R_3$: If Petal width is $L_5$ and Petal length is $L_{13}$ then class is Virginica.

The most sensitive parameters of our feature selection algorithm, which also affect the fuzzy rule-based classifier, include: $n'$ as the number of desired features, and $K'$ as the number of fuzzy sets on each feature. To examine the sensitivity of our methods on these parameters, two data sets are used: Wine and Tiger with low and high dimensions, respectively. For this purpose, three distinct values of $n'$ (3, 4 and 5) versus all possible values of $K'$ (1, 2, …, 14) are studied. In this regard, $n'$ best features with $K'$ best fuzzy sets, reported by feature selection algorithm, are used to design the fuzzy classifiers. The accuracy, in (11), of these classifiers are computed by using the training data for test, also. Figure 3 depicts these accuracies for Wine and Tiger data sets. In both data, the number of features is not determinant, at least for 3, 4 and 5 features. This also happens for number of fuzzy sets in each feature, except when weak fuzzy sets are included. Clearly, using only the best fuzzy set is sufficient since the obtained accuracies are not influenced by more fuzzy sets. However, in the coming

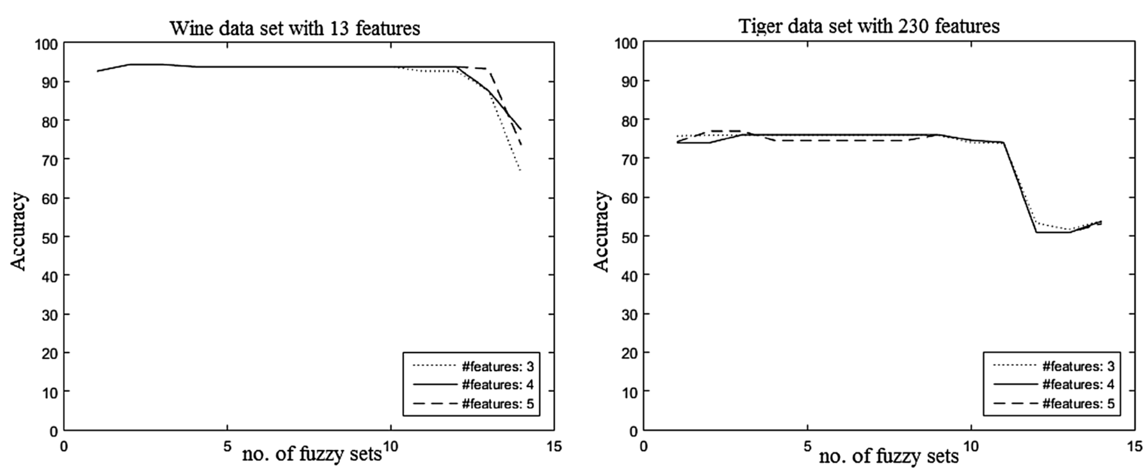


**Fig. 3** Effects of number of features ($n'$) and fuzzy sets ($K'$) on accuracy of designed classifiers

**Table 4** Computational cost (in sec) of four feature selection methods

| Data set | Fischer | mRMR | Entropy-based | Proposed method |
|---|---|---|---|---|
| Bupa | 7 | 2 | 2 | **2** |
| Cancer | 24 | 7 | **6** | 7 |
| Ecoli | 30 | 7 | 17 | **6** |
| Glass | 19 | 6 | 4 | **3** |
| Image | 67 | 14 | 5 | **5** |
| Ionosphere | 325 | 50 | 32 | **21** |
| Pima | 21 | 252 | 5 | **5** |
| Segment | 616 | 408 | **71** | 93 |
| Sonar | 279 | 148 | **35** | 37 |
| Vehicle | 420 | 40 | 44 | **25** |
| Vowel | 172 | 407 | 41 | **38** |
| Wine | 14 | 4 | 3 | **2** |
| Yeast | 150 | **27** | 68 | 39 |
| Cnae9 | 11,550 | **2079** | 5236 | 3003 |
| Coil2000 | 692 | 604 | 74 | **52** |
| Fox | 437 | 127 | **109** | 120 |
| Musk | 312 | 91 | **78** | 86 |
| Secom | 25,350 | 3900 | 2496 | **1638** |
| Tiger | 8775 | 1350 | 864 | **567** |
| Average | 2592.6 | 501.2 | 483.7 | **302.6** |

**Table 5** Classification accuracy obtained using different feature selection methods

| Data set | mRMR | Fischer | Entropy-based | Proposed method |
|---|---|---|---|---|
| Bupa | **59.46** | 58.15 | 61.68 | 58.40 |
| Cancer | 94.76 | 94.12 | 93.08 | **94.49** |
| Ecoli | 75.13 | **78.38** | 63.20 | 76.27 |
| Glass | 46.60 | **63.12** | 45.97 | 55.94 |
| Image | 69.14 | 72.57 | 11.43 | **73.23** |
| Ionosphere | 64.22 | 64.02 | 78.59 | **85.89** |
| Iris | 92.14 | **94.67** | 72.67 | 94.27 |
| Pima | **74.46** | 73.78 | 66.52 | 73.21 |
| Segment | 72.13 | 72.57 | 18.37 | **72.69** |
| Sonar | 67.98 | **69.44** | 66.00 | 69.09 |
| Vehicle | 47.69 | 47.51 | 49.08 | **52.95** |
| Vowel | 56.38 | **57.43** | 36.75 | 54.48 |
| Wine | **92.86** | 90.54 | 75.17 | 91.60 |
| Yeast | 67.76 | 69.48 | 60.44 | **70.02** |
| Coil2000 | 94.03 | 94.03 | 94.03 | **94.03** |
| Musk | 56.52 | 57.18 | 56.54 | **57.19** |
| Fox | 52.12 | 51.48 | 50.53 | **57.55** |
| Tiger | 65.98 | 60.69 | 55.80 | **70.49** |
| Secom | 93.19 | **93.36** | 93.36 | 93.20 |
| Cnae9 | 63.28 | 64.94 | 57.37 | **65.80** |
| Average | 70.29 | 71.37 | 60.33 | **73.04** |

**Table 6** *p* Value of paired *t* test on the classification accuracy of methods

| | mRMR | Fischer | Entropy-based |
|---|---|---|---|
| Proposed method | 0.0154 | 0.1096 | 0.0016 |

experiments, the number of features and fuzzy sets are set to 4.

The experimental results are studied in two subsections. First, our fuzzy feature selection method is evaluated. Next, the proposed method for designing fuzzy rule-based classifiers is discussed. All methods are implemented in MATLAB R2014 and are run on a Core i5, 3.1-GHz CPU with 4 GB of memory in Windows 7.

To compare the different approaches in a formal and efficient manner, the five times of tenfold cross-validation (5–10CV) testing method is used. In this method, each data set is randomly divided into ten subsets of the same size. Nine subsets are used for training and the tenth subset is used for test. The same training and testing procedure is also performed nine times after exchanging the role of each subset so that all subsets are used as test patterns once. Since the error rate on test patterns depends on the initial division of the data set, the 10CV is iterated five times using different divisions of the data set and the average accuracy is reported.

### 5.1 Examining our fuzzy feature selection approach

In this part, our fuzzy feature selection algorithm is compared with mRMR method (Peng et al. 2005), Fischer criterion (Fisher 1936), and a fuzzy method on the basis of fuzzy entropy and similarity measure (Luukka 2011). The mRMR method is based on maximizing the relevancy and minimizing the redundancy between the features using mutual information. In Fischer criterion, the ratio of traces of within-class and between-class scatter matrices in each dimension is the basis of ranking the features. Table 4 examines the scalability of methods in selecting four features for each data set. The computational cost of methods is stated in terms of CPU time. As shown in boldfaces, the CPU cost of our proposed algorithm is less than other methods in most of data sets and also in average.

Using the obtained features by each method, the approach in Ishibuchi and Yamamoto (2004) for designing fuzzy classifiers is employed to examine the influence of selected features on the classification accuracy and so the effectiveness of each method in feature selection. But since our proposed method selects the features class-specifically, an ensemble of classifiers for each class (Pineda-Bautista et al. 2011) is applied in this case. Using the features

**Table 7** Comparing the performance of our proposed method versus Ishibuchi method

| Data set | Accuracy (%) | | CPU time (s) | | Length of rules ($n'$) | | No. of rules ($\|FRB\|$) | |
|---|---|---|---|---|---|---|---|---|
| | Ishibuchi method | Proposed method | Ishibuchi method | Proposed method | Ishibuchi method | Proposed method | Ishibuchi method | Proposed method |
| Ecoli | 68.92 | 76.79 | 1016 | 79 | 1.8 | 1.2 | 8.4 | 11.1 |
| Glass | 42.34 | 48.48 | 541 | 77 | 1.8 | 1.6 | 7.6 | 9.0 |
| Ionosphere | 73.07 | 86.72 | 146,420 | 60 | 1.7 | 1.9 | 3.8 | 6.1 |
| Pima | 67.14 | 74.13 | 221 | 78 | 1.0 | 1.5 | 2.3 | 3.7 |
| Segment | 67.29 | 70.57 | 22,811 | 740 | 1.8 | 2.2 | 12.2 | 7.4 |
| Sonar | 71.45 | 73.84 | 562,319 | 65 | 1.3 | 2.1 | 5.2 | 5.7 |
| Thyroid | 77.74 | 92.69 | 115 | 38 | 1.0 | 2.3 | 3.9 | 8.2 |
| Yeast | 60.65 | 72.51 | 4695 | 228 | 1.8 | 1.6 | 8.0 | 9.3 |
| Average | 66.06 | **74.47** | 92,267 | **170** | **1.5** | 1.8 | **6.4** | 7.6 |
| Bupa | 57.73 | 56.88 | 120 | 28 | 1.4 | 1.1 | 2.9 | 5.2 |
| Cancer | 94.55 | 96.08 | 1192 | 84 | 1.0 | 2.2 | 2.9 | 5.9 |
| Image | 69.81 | 70.95 | 5741 | 82 | 1.7 | 1.7 | 8.6 | 11.3 |
| Iris | 96.13 | 96.13 | 114 | 20 | 1.0 | 1.6 | 4.1 | 6.4 |
| Vehicle | 46.84 | 48.64 | 6768 | 204 | 1.0 | 1.2 | 9.0 | 9.1 |
| Wine | 93.25 | 94.62 | 1195 | 40 | 1.1 | 2.1 | 5.2 | 6.5 |
| Average | 76.39 | **77.22** | 2522 | **76** | **1.2** | 1.7 | **5.5** | 7.4 |

obtained by four feature selection methods, the performance of designed fuzzy classifiers, in terms of accuracy in (11), is compared in Table 5. According to these accuracies, the performance of our approach (in selecting discriminative features and suitable fuzzy sets) is much better than the entropy-based method and comparable to mRMR and Fischer methods.

To justify our claim statistically, the $t$ test (Kreyszig 1970) is examined on the null hypothesis that the classification accuracy of our proposed method in 14 + 6 data sets is not better than the others. The $p$ value of paired comparisons with $\alpha = 0.05$ are reported in Table 6 where its small values cast doubt on the validity of the null hypothesis. Clearly, the differences are statistically significant and the performance of our method is considerably better than the entropy-based and mRMR methods but not than Fischer criterion.

### 5.2 Performance of our method in designing fuzzy rule-based classifiers

In this subsection, the performance of our proposed method for generating fuzzy classification rules is compared with a well-known method; proposed by Ishibuchi and Yamamoto (2004). Because of its rule-length constraints, only fuzzy rules with the length of at most two are generated. Also, since it needs an evaluation criterion, the product of confidence and support (Ishibuchi and Yamamoto 2004) is used for this purpose. To construct the final rule base, the proposed method, in this paper, is also employed for Ishibuchi approach. Additionally, the number of rules per class ($Q$ in algorithm) is set to 5 for both methods. The performance of our algorithm versus Ishibuchi method is compared in Table 7 in terms of classification accuracy in (11), computational cost, number of rules in the final rule base, $|FRB|$, and length of rules, $n'$. In this table, only low- and moderate-dimensional data sets are used for comparisons because of mentioned constraint of Ishibuchi's approach.

The data sets in this table are grouped in two categories because of the distinct performances. In the first group, the classification accuracy of our algorithm is significantly better than Ishibuchi method, about 8 % in average. Even in second group, our accuracies are almost better. This is because of selecting discriminative features for each class and suitable fuzzy sets for each feature. Moreover, the proposed criterion for rule evaluation puts the more accurate rules at top of the ranking list. The reported computational costs clearly justify the scalability of our approach in generating a compact set of short fuzzy rules for high-dimensional and/or large data. Moreover, the size of rule bases and the length of rules in proposed method are near to, but not as good as, those in Ishibuchi approach. This is because, the length of fuzzy rules in his approach are restricted to two while in our method the rules are allowed to be longer (indeed, till the number of selected features; at most 4 in these experiments). So, the generated rules of Ishibuchi are more general than our rules and therefore, our method must generate more rules to cover the same subspace of the problem.

**Table 8** Classification accuracy of our method against SGERD and FARC-HD

| Data set | Proposed method | SGERD | FARC-HD |
|---|---|---|---|
| Bupa | 56.88 | 54.13 | **60.85** |
| Cancer | 96.08 | **96.20** | 96.15 |
| Ecoli | 76.79 | 70.94 | **77.97** |
| Glass | 48.48 | 52.86 | **62.53** |
| Image | 70.95 | **80.46** | 78.76 |
| Ionosphere | 86.72 | 88.43 | **91.05** |
| Iris | **96.13** | 94.27 | 96.00 |
| Pima | **74.13** | 73.68 | 73.65 |
| Segment | 70.57 | 78.46 | **82.73** |
| Sonar | 73.84 | 70.50 | **76.35** |
| Vehicle | 48.64 | 47.77 | **53.60** |
| Wine | 94.62 | **95.03** | 92.63 |
| Cnae9 | **65.80** | 54.39 | 62.53 |
| Coil2000 | 93.64 | **93.72** | – |
| Fox | 100.00 | 99.79 | 100.00 |
| Musk | 65.89 | 62.78 | **66.20** |
| Secom | 93.00 | 93.27 | **93.36** |
| Tiger | 74.11 | **76.80** | 74.40 |
| Average | 77.02 | 76.86 | **78.75** |

As the last experiments, the effectiveness of our method against some state-of-the-art fuzzy classifier design schemes is compared. In this regard, two recently-proposed and efficient methods in fuzzy classifier design are used. The first one is SGERD (Mansoori et al. 2008), a steady-state genetic algorithm for extracting fuzzy classification rules from data. This method is fast and scalable with acceptable accuracy. The second algorithm is FARC-HD (Alcala-Fdez et al. 2011a), a fuzzy association rule-based classification model for high-dimensional problems with genetic rule selection and lateral tuning. FARC-HD is efficient and accurate, because of its goodness in rule selection and lateral tuning.

However, the computational complexity of these methods should be managed, especially when are applied on moderate- and high-dimensional data. For this purpose, first our proposed feature selection method is applied on each data set and then, the four best features are used by SGERD and FARC-HD. In these experiments, the implementation of algorithms in Keel data-mining software tool (Alcala-Fdez et al. 2011b) are used while their required parameters are set to default values.

Table 8 compares their classification accuracy in (11) against our method. Clearly, the performance of designed classifiers for some data sets is weak because of inadequacy of features. By selecting more than four features for these data sets, their classification rate would hopefully be increased.

According to the results in this table, the performance of our method is comparable to SGERD and FARC-HD, though the latter one is more accurate, both in average and in most of the data sets. This is because of its lateral tuning and the rules weight usage.

## 6 Conclusion

In this paper, we proposed a novel and fast method for fuzzy feature selection to choose more relevant features; those which can distinguish the distinct classes well. Our method uses the membership degree of positive and negative patterns in the fuzzy sets in order to compute the relevancy of features to the classes. The selected features and their effective fuzzy sets were then used in designing fuzzy rule-based classifiers. In order to evaluate the initially generated candidate rules, a new criterion was also proposed to measure the class-discrimination ability of each fuzzy rule.

The experimental results showed that our feature selection method is fast and scalable to be applied on high-dimensional data. By using just a few of these features, our approach for designing fuzzy rule-based classifiers could generate accurate and interpretable rule bases. In future works, we should develop a fuzzy feature selection approach which can also detect the redundant features.

## References


Alcala-Fdez J, Alcala R, Herrera F (2011a) A fuzzy association rule-based classification model for high-dimensional problems with genetic rule selection and lateral tuning. IEEE Trans Fuzzy Syst 19(5):857–872

Alcala-Fdez J, Fernandez A, Luengo J, Derrac J, Garcia S, Sanchez L, Herrera F (2011b) Keel data-mining software tool: data set repository, integration of algorithms and experimental analysis framework. J Multiple Valued Logic Soft Comput 17(2–3):255–287

Almaksour A, Anquetil E (2011) Improving premise structure in evolving Takagi-Sugeno neuro-fuzzy classifiers. Evol Syst 2:25–33

Angelov P, Lughofer E, Zhou X (2008) Evolving fuzzy classifiers using different model architectures. Fuzzy Sets Syst 159(23):3160–3182

Asuncion A, Newman DJ (2007) UCI machine learning repository. Department of Information and Computer science, University of California, Irvine

Bouchachia A, Mittermeir R (2006) Towards incremental fuzzy classifiers. Soft Comput 11(2):193–207

Casillas J, Cordon O, Del Jesus MJ, Herrera F (2001) Genetic feature selection in a fuzzy rule-based classification system learning process for high dimensional problems. Inf Sci 136:135–157

Cassillas J, Cordon O, Del Jesus MJ, Herrera F (2001) Genetic feature selection in a fuzzy rule-based classification system learning process for high-dimensional problems. Int J Inf Sci 136:135–157

Chakraborty D, Pal NR (2004) A neuro-fuzzy scheme for simultaneous feature selection and fuzzy rule-based classification. IEEE Trans Neural Netw 15:110–123

Chiu S (1994) Fuzzy model identification based on cluster estimation. J Intell Fuzzy Syst 2:276–278

Cordon O, Del Jesus MJ, Herrera F, Lozano M (1999) MOGUL: a methodology to obtain genetic fuzzy rule based systems under the iterative rule learning approach. Int J Intell Syst 14(11):1123–1143

Estevez PA, Tesmer M, Perez CA, Zurada JM (2009) Normalized mutual information feature selection. IEEE Trans Neural Netw 20(2):189–201

Fisher RA (1936) The use of multiple measurements in taxonomic problems. Ann Eugen 7:179–188

Gacto MJ, Alcala R, Herrera F (2011) Interpretability of linguistic fuzzy rule-based systems: an overview of interpretability measures. Inf Sci 181(20):4340–4360

Guyon I, Elisseeff A (2003) An introduction to variable and feature selection. J Mach Learn Res 3:1157–1182

Halgamuge S, Glesner M (1994) Neural networks in designing fuzzy systems for real world applications. Fuzzy Sets Syst 65(1):1–12

Iglesias JA, Angelov P, Ledezma A, Sanchis A (2010) Evolving classification of agent's behaviors: a general approach. Evol Syst 1(3):161–172

Ishibuchi H, Murata T (1997) Minimizing the fuzzy rule base and maximizing its performance by a multi-objective genetic algorithm. In: Proceedings of 6th FUZZ-IEEE, pp 259–264

Ishibuchi H, Nakashima T (2001) Effect of rule weights in fuzzy rule-based classification systems. IEEE Trans Fuzzy Syst 9(4):506–515

Ishibuchi H, Yamamoto T (2004) Comparison of heuristic criteria for fuzzy rule selection in classification problems. Fuzzy Optim Decis Making 3(2):119–139

Ishibuchi H, Nakashima T, Morisawa T (1999) Voting in fuzzy rule-based systems for pattern classification problems. Fuzzy Sets Syst 103(2):223–238

Kreyszig E (1970) Introductory mathematical statistics. John Wiley, New York

Lee HM, Chen CM, Chen JM, Jou YL (2001) An efficient fuzzy classifier with feature selection based on fuzzy entropy. IEEE Trans Syst Man Cybern Part B Cybern 31(3):426–432

Lughofer E (2011) On-line incremental feature weighting in evolving fuzzy classifiers. Fuzzy Sets Syst 163(1):1–23

Lughofer E, Buchtala O (2013) reliable all-pairs evolving fuzzy classifiers. IEEE Trans Fuzzy Syst 21(4):625–641

Lughofer E, Bouchot J-L, Shaker A (2011) On-line elimination of local redundancies in evolving fuzzy systems. Evol Syst 2(3):165–187

Luukka P (2011) Feature selection using fuzzy entropy measures with similarity classifier. Expert Syst Appl 38:4600–4607

Mansoori EG, Zolghadri MJ, Katebi SD (2007) A weighting function for improving fuzzy classification systems performance. Fuzzy Sets Syst 158(5):583–591

Mansoori EG, Zolghadri MJ, Katebi SD (2008) SGERD: a steady-state genetic algorithm for extracting fuzzy classification rules from data. IEEE Trans Fuzzy Syst 16(4):1061–1071

Marin-Blazquez JG, Shen Q (2002) From approximative to descriptive fuzzy classifiers. IEEE Trans Fuzzy Syst 10(4):484–497

Nauck D, Kruse R (1997) A neuro-fuzzy method to learn fuzzy classification rules from data. Fuzzy Sets Syst 89(3):277–288

Pedrycz W (1994) Why triangular membership functions? Fuzzy Sets Syst 64(1):21–30

Peng H, Long F, Ding C (2005) Feature selection based on mutual information: criteria of max-dependency, max-relevance, and min-redundancy. IEEE Trans Pattern Anal Mach Intell 27(8):1226–1238

Pineda-Bautista BB, Carrasco-Ochoa JA, Martinez-Trinidad JF (2011) General framework for class-specific feature selection. Expert Syst Appl 38:10018–10024

Rehm F, Klawonn F, Kruse R (2007) Visualization of fuzzy classifiers. Int J Uncertain Fuzziness Knowl Based Syst 15(5):615–624

Roubos H, Setnes M (2000) Compact fuzzy models through complexity reduction and evolutionary optimization. Proc Ninth IEEE Int Conf Fuzzy Syst 2:762–767

Setnes M, Babuska R, Kaymak U, van Nauta-Lemke HR (1998) Similarity measures in fuzzy rule base simplification. IEEE Trans Syst Man Cybern Part B Cybern 28:376–386

Shie JD, Chen SM (2007) Feature subset selection based on fuzzy entropy measures for handling classification problems. Appl Intell 28:69–82

Tuv E, Borisov A, Runger G, Torkkola K (2009) Feature selection with ensembles, artificial variables, and redundancy elimination. J Mach Learn Res 10:1341–1366

Yang J, Honavar V (1998) Feature subset selection using a genetic algorithm. IEEE Intell Syst 13(2):44–49