A Fuzzy Bayesian Classifier with Learned Mahalanobis Distance

Necla Kayaalp,[*] Guvenc Arslan[†]
Department of Mathematics, Izmir University of Economics, 35330, Izmir, Turkey

Recent developments show that naive Bayesian classifier (NBC) performs significantly better in applications, although it is based on the assumption that all attributes are independent of each other. However, in the NBC each variable has a finite number of values, which means that in large data sets NBC may not be so effective in classifications. For example, variables may take continuous values. To overcome this issue, many researchers used fuzzy naive Bayesian classification for partitioning the continuous values. On the other hand, the choice of the distance function is an important subject that should be taken into consideration in fuzzy partitioning or clustering. In this study, a new fuzzy Bayes classifier is proposed for numerical attributes without the independency assumption. To get high accuracy in classification, membership functions are constructed by using the fuzzy C-means clustering (FCM). The main objective of using FCM is to obtain membership functions directly from the data set instead of consulting to an expert. The proposed method is demonstrated on the basis of two well-known data sets from the literature, which consist of numerical attributes only. The results show that the proposed the fuzzy Bayes classification is at least comparable to other methods. © 2014 Wiley Periodicals, Inc.

1. INTRODUCTION

Recent developments and research show that a more general concept of a distance function in learning algorithms provides improved performance over classically used distance functions such as the Euclidean distance (see, e.g., Pekelska et al.,[1] and Khemchandani et al.[2]). For example, the choice of an appropriate dissimilarity measure is an important and crucial step in clustering algorithms. On the other hand, since cluster analysis is typically used in an exploratory context, it is usually not known in advance which dissimilarity measure is best suited for the data at hand. Hence, the choice is generally made by an expert, based on some prior information or experience, if available. Another reason why a more general concept of a distance function should be considered is that the appropriate dissimilarity measure to be used may not actually be a metric function as is usually the the case in classical approaches. Li and Lu,[3] for example, note that in computer

[*] Author to whom all correspondence should be addressed; e-mail: necla.kayaalp@ieu.edu.tr.
[†] e-mail: guvenc.arslan@ieu.edu.tr

INTERNATIONAL JOURNAL OF INTELLIGENT SYSTEMS, VOL. 29, 713–726 (2014)

View this article online at wileyonlinelibrary.com. • DOI 10.1002/int.21659

vision the traditional Euclidean distance cannot reflect the real distance between images.

In the literature, one can find recent studies on distance metric learning. One particular view is that one should consider side information such as specifying which pairs are similar and which are dissimilar. Xiang et al.,[4] for example, developed an algorithm to learn a Mahalanobis distance metric by supplying prior knowledge in terms of similar and dissimilar data pairs, which are called must-links and cannot links, respectively. The learned Mahalanobis distance then can be used in a clustering or classification algorithm. It is expected that the learned distance function will improve the performance of the algorithm.

The naive Bayesian classifier (NBC) is a popular and well-known classifier used by many researchers. It is based on the conditional independence assumption of the features.[6] This assumption simplifies the calculations in the method. Despite this strong assumption, it gives surprisingly good results. One can find many studies and extensions related to NBC. For example, Harry and Sheng,[7] used weights as powers of the conditional probabilities. Yager[8] extended the NBC by using ordered weighted averaging. In addition, one may also find some studies that combine fuzzy set theory with NBC. Störr,[9] for example, used fuzzy membership functions instead of prior probabilities. Tang et al.,[10] on the other hand, approached the classification problem by using fuzzy clustering as an intermediate step.

In this paper, we consider a new classifier by applying the fuzzy c-means (FCM) clustering algorithm with a learned Mahalanobis distance. The proposed classifier may be considered as another interpretation of Bayes' theorem using fuzzy numbers. The algorithm for learning the Mahalanobis distance is almost the same as described by Xiang et al.[4] One of the basic differences in our method is that the must-links and cannot-links are obtained directly from the data set as opposed to the approach of Xiang et al.[4] This is achieved by using similarities between the data points. The classification is achieved by using fuzzy membership functions, which are constructed from the obtained clusters by using the learned Mahalanobis distance.

2. PRELIMINARIES AND BASIC NOTATION

Since in clustering and classification problems, objects/examples are described by attributes we will use the notion of a *data table* to describe a general data set. A data table is also referred to as an information system and consists of a 4-tuple $\langle U, A, V, f\rangle$, where U is a finite set of objects and $A = \{a_1, a_2, \ldots, a_m\}$ is a finite set of attributes. The domain of an attribute $a \in A$ is denoted by V_a and $V = \bigcup_{a\in A} V_a$. The function f is a total function such that $f(x, a) \in V_a$ for each $a \in A$, $x \in U$, and it is called an information function. If the set of attributes A is divided into *condition* attributes ($C \neq \phi$) and *decision* attributes ($D \neq \phi$), then the data table is called a *decision table* (see Greco et al.[5]).

An important step in the proposed method is the construction of the must-link and cannot-link sets. This will be achieved by using similarities between examples of the data set. The following definitions, which are given in Chen and Wang[11] and Greco et al.,[5] will be used to obtain the sets of must-links and cannot-links.

Definition 2.1. *The similarity value between two examples $x, y \in U$ with respect to an attribute $a \in A$ is defined as*

$$sim_a(x_i, x_j) = 1 - \frac{\left|f(x_i, a) - f(x_j, a)\right|}{\max(a) - \min(a)},$$

if a is a numerical attribute and as

$$sim_a(x_i, x_j) = \begin{cases} \dfrac{1}{card\{a\}}, & if \quad f(x_i, a) \neq f(x_j, a), \\ 1, & otherwise \end{cases}$$

if a is a nominal attribute.

Definition 2.2. *The similarity value between two examples $x, y \in U$ with respect to an attribute set $B \subseteq A$ is defined as*

$$sim_B(x_i, x_j) = \sum_{a \in B} w_a sim_a(x_i, x_j)$$

where the weight w_a corresponds to the attribute $a \in B$.

Xiang et al.[4] used entropy to determine the weights in Definition 2 in their clustering algorithm. In this study, we use class labels to determine the weights w_a, for $a \in B$. The details will be explained in the next section.

The set of must-links is constructed by determining a threshold value, which will be used to decide whether two examples are in some sense *indistinguishable*. Hence, we also give the following definition of *indiscernibility*, which is a basic concept in rough set theory:

Definition 2.3. *The* indiscernibility *relation IR_B with confidence level t, with respect to an attribute set $B \subseteq A$ is defined as*

$$IR_B(t) = \{(x_i, x_j) \in U \times U : sim_B(x_i, x_j) \geq t\}$$

where t is the threshold value for the similarity relation (see Chen and Wang[11]).

Using these definitions, it is possible to construct the must-link and cannot-link sets by using an appropriate threshold value t.

A must-link set, with respect to an attribute set $B \subseteq A$, can now be defined by $S_B(t) = IR_B(t)$. In a similar way, a cannot-link set, with respect to an attribute set $B \subseteq A$, is defined by

$$D_B(\epsilon) = \{(x_i, x_j) | \mathrm{sim}_B(x_i, x_j) \leq \epsilon\}$$

As can be seen from these definitions, $S_B(t)$ is a set that contains points that are considered to be definitely in a same class whereas $D_B(\epsilon)$ is a set that contains points that are considered definitely to be in different classes. For convenience, we will write $S(t)$ and $D(\epsilon)$ to denote the must-link and cannot-link sets with respect to all attributes considered.

Remark 2.1. We note that the must-link and cannot-link sets, as defined above, are binary relations on U, which are reflexive and symmetric but not transitive.

3. THE PROPOSED CLASSIFIER

The NBC is a well-known and widely used classifier. The classifier proposed in this paper, in some sense, is based on a similar idea as in the NBC. The main idea is to learn a Mahalanobis distance (via must-links and cannot-links) for clustering the samples and to use fuzzy numbers to achieve classification. We note that the assumption of conditional independence of features is no longer necessary. It will be seen that the formulation of this new classifier may actually be considered as another interpretation of Bayes' theorem.

Bayes theory is a kind of probability theory providing a mathematical framework for making inference with probabilities, and Bayes' theorem is a statement in conditional probabilities such that prior probabilities are mapped into posterior probabilities by using class label information or outcome of classification events. Generally, prior probabilities are obtained by frequencies of attributes or knowledge of an expert that may not yield high accuracy in classification algorithms. However, prior probabilities may be generated objectively without consulting an expert. In this study, it is shown that this prior knowledge can be derived from the data set. Then, posterior information is obtained based on this prior knowledge. To sum up, the logic behind Bayes' theorem is to get posterior knowledge by using prior knowledge. In the proposed method, one can see that Bayesian logic is used implicitly such that to classify a new example class label information is used. To be more precise, in the proposed method partition of samples into clusters is achieved as in the NBC (see Figure 1). Then, the basic principle as in Bayes' theorem is applied to find conditional membership functions. More formally, conditional probabilities and conditional membership functions are defined as

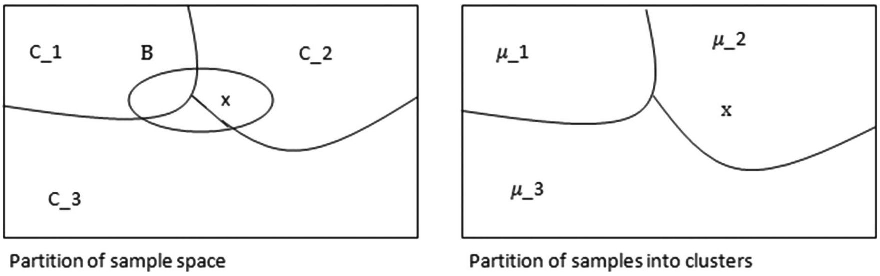
$$p(C_j|\mathbf{x}) = \frac{p(\mathbf{x}|C_j)p(C_j)}{\sum_i p(\mathbf{x}|C_i)p(C_i)}$$

and

$$\mu(C_j|\mathbf{x}) = \frac{\mu_l(C_j)}{\max_i \mu_i(\mathbf{x})}, \quad l = \text{argmax}_{1 \le i \le k} \{\mu_i(x)\},$$

respectively.

The proposed classifier consists of two main steps. First, fuzzy membership functions for the clusterings of the data set, obtained by applying fuzzy c-means with a learned Mahalanobis distance, are constructed. Second, using these membership functions, the classification is achieved (see Figure 2). The basic steps are summarized as follows:



Partition of sample space

Partition of samples into clusters

Figure 1. The partition of the sample space within probabilities and fuzzy approaches.

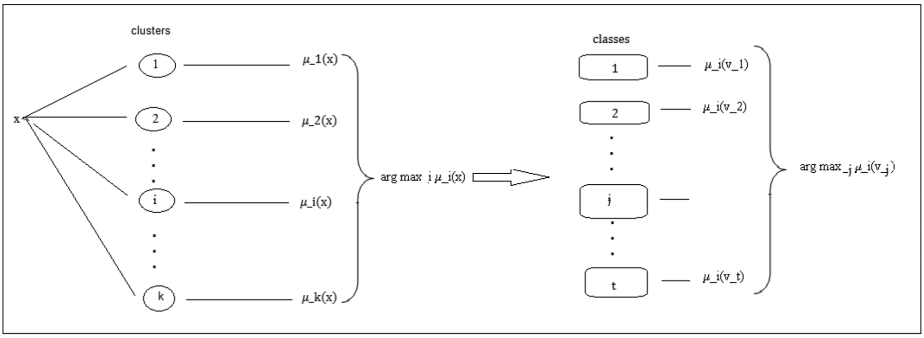


Figure 2. The process for classification of a new example.

(1) Construct the fuzzy membership functions.
   (a) Construct the sets for must-links and cannot-links.
   (b) Learn a Mahalanobis distance to be used in the next step.
   (c) Apply fuzzy c-means algorithm and obtain clusters.
   (d) Construct fuzzy membership functions for the clusters.
(2) Apply the classification method.
   (a) Find cluster with the highest membership for a new example x_0.
   (b) By evaluating memberships of class centers, determine the class of x_0.

In the first step, using the indiscernibility relation given in Definition 3 with an appropriate threshold value, the sets of pairs of must-links and cannot-links are determined directly from the data set. To achieve this, the weights w_a, $a \in C$, are determined by using class labels as follows: Suppose that there are t classes in the data set. For $a \in C$, let

$$A_i(a) = \{x \in U \,|\min(C_i(a)) \leq f(x,a) \leq \max(C_i(a))\},$$

where $C_i(a)$ is the set of values for attribute a belonging to class i, $1 \leq i \leq t$. Denoting by

$$B_j(a) = A_j(a) \backslash \bigcup_{i=1(i \neq j)}^{t} A_i(a),$$

the weights w_a are defined as

$$w_a = \frac{\sum_{i=1}^{t} s\left(B_i(a)\right)}{s(U)}$$

where $S(B)$ denotes the number of elements in set B.

However, weights are normalized to see the impact of that attribute with respect to each class. Therefore,

$$w_a^* = \frac{w_a}{\sum_a w_a}$$

To explain the proposed method for computing weights of each attribute, let us consider an example. A sample data set consisting 40 examples with two attributes one of which is the class attribute is chosen from a well-known data set, the Fisher Iris data set. This sample data set includes the first 20 examples from the class Setosa and the first 20 examples from the class Virginica. It is seen that the weights of each attribute are computed as $w_1^* = 23/35$ and $w_2^* = 12/35$ (see Figure 3)

In the second step, using the sets of must-links and cannot-links as side information, a Mahalanobis distance is learned by applying the same steps as described by Xiang et al.[4] In other words, Xiang et al.'s algorithm is used to find the optimum matrix W^* for $A = W^*(W^*)^T$ to be used as a Mahalanobis distance. For this purpose Xiang et al.[4] used a transformation such that $y = W^T x$, where $W \in \mathbb{R}^{n \times d}$, with $d \leq n$. Based on this transformation, the sum of the squared distances of the point pairs in $S(t)$ is defined as

$$d_w = \sum_{\left(x_i, x_j\right) \in S} \left(W^T x_i - W^T x_j\right)^T \left(W^T x_i - W^T x_j\right) = tr\left(W^T S_w W\right),$$

where tr is the trace operator and S_w, the covariance matrix of the point pairs in $S(t)$, is calculated as

$$S_w = \sum_{(x_i, x_j) \in S} (x_i - x_j)(x_i - x_j)^T$$

Similarly, for the point pairs in $D(\epsilon)$, we have

$$d_b = tr\left(W^T S_b W\right),$$

where S_b is the covariance matrix of the point pairs in $D(\epsilon)$ is calculated as

$$S_b = \sum_{\left(x_i, x_j\right) \in D} \left(x_i - x_j\right)\left(x_i - x_j\right)^T.$$

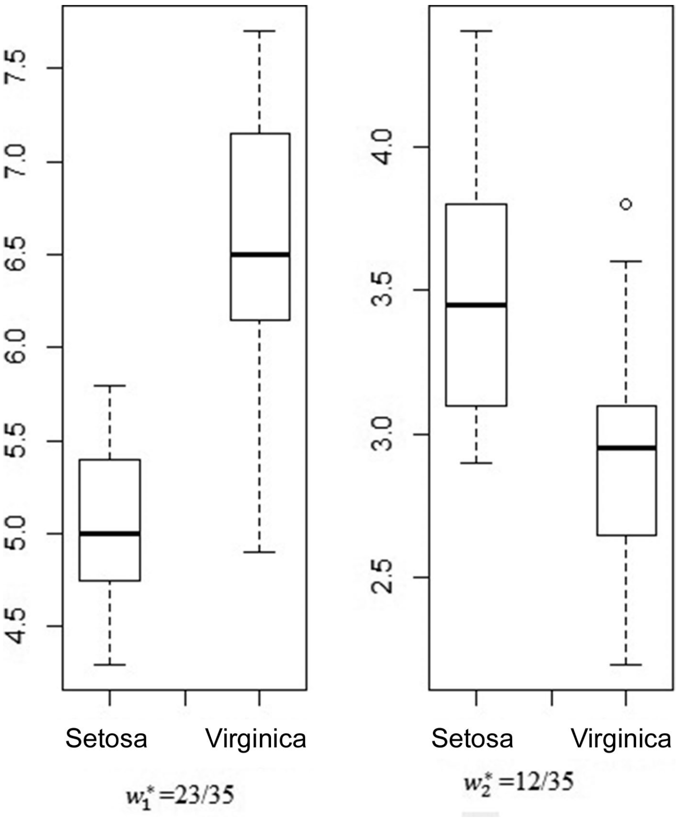


Figure 3. Weights of two attributes for the chosen data set.

Since d_w and d_b represent the sum of squared distances between point pairs in must-links and cannot links, respectively, the optimal matrix W^* can be calculated as

$$W^* = \text{argmax}_{\{W^T W = I\}} \frac{tr(W^T S_b W)}{tr(W^T S_w W)},$$

where I is an identity matrix and the constraint $W^T W = I$ is given in order not to have degenerate solutions. The important point here is that W cannot be a square matrix when $d < n$. In that case, A is defined as follows:

$$A = \begin{cases} (W^*(W^*)^T, & \text{if} \quad d < n \\ I, & \text{if} \quad d = n \end{cases}$$

It is crucial to note that for different values of d, the accuracy rate of clustering and classification may change.

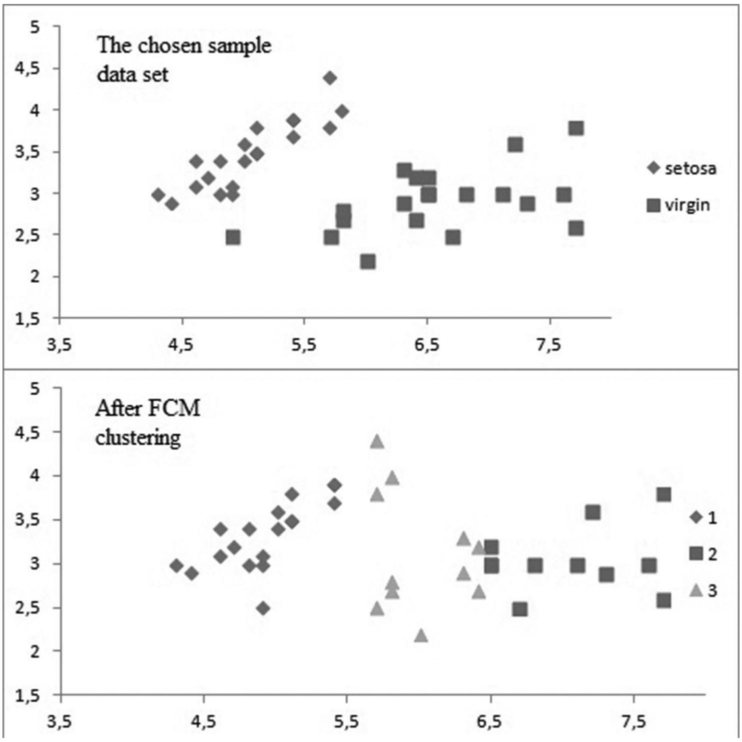


Figure 4. Chosen sample data set.

In the third step, the learned Mahalanobis distance is used to apply the fuzzy c-means algorithm to obtain clusters, which will be used to construct the fuzzy membership functions as defined in Tang et al.[10] These fuzzy membership functions, denoted by $\mu_1, \ \mu_2, \ \ldots, \ \mu_k$, will then be used in the classification process. We note that for any $x \in U$, $\mu_i(x), \ 1 \leq i \leq k$ is defined by

$$\mu_i(x) = \sum_{a \in C} w_a \mu_{i,a} \left(f(x, a) \right),$$

where $\mu_{i,a}$ denotes the component of the membership function for the ith cluster corresponding to attribute $a \in C$. The weights w_a in the above formula are determined using weight formula, which is defined in the proposed method again, but this time with clusters instead of classes.

To illustrate the steps we explained so far, consider the chosen sample data set used before in computing the weights of each attribute. The resulting clustering for $c = 3$ clusters obtained by applying the described steps is shown together with their centers (v_1, v_2, v_3) in Figure 4.

The fuzzy membership functions for the three clusters are computed as follows:

$$\mu_{1,a_1}(x_j) = \begin{cases} 1, & \text{if} \quad x_j \leq 4.956 \\ \dfrac{5.939 - x_j}{5.939 - 4.956}, & \text{if} \quad 4.956 \leq x_j \leq 5.939 \\ 0, & \text{if} \quad x_j > 5.939 \end{cases}$$

$$\mu_{2,a_2}(x_j) = \begin{cases} 1, & \text{if} \quad x_j \leq 3.06 \\ \dfrac{3.124 - x_j}{3.124 - 3.06}, & \text{if} \quad 3.06 \leq x_j \leq 3.124 \\ 0, & \text{if} \quad x_j > 3.124 \end{cases}$$

$$\mu_{3,a_1}(x_j) = \begin{cases} 0, & \text{if} \quad x_j \leq 4.956 \\ \dfrac{x_j - 4.956}{5.939 - 4.956}, & \text{if} \quad 4.956 \leq x_j \leq 5.939 \\ \dfrac{6.936 - x_j}{6.936 - 5.939}, & \text{if} \quad 5.939 < x_j \leq 6.936 \\ 0, & \text{if} \quad x_j > 6.936 \end{cases}$$

$$\mu_{3,a_2}(x_j) = \begin{cases} 0, & \text{if} \quad x_j \leq 3.06 \\ \dfrac{x_j - 3.06}{3.124 - 3.06}, & \text{if} \quad 3.06 \leq x_j \leq 3.124 \\ \dfrac{3.27 - x_j}{3.27 - 3.124}, & \text{if} \quad 3.124 < x_j \leq 3.27 \\ 0, & \text{if} \quad x_j > 3.27 \end{cases}$$

$$\mu_{2,a_1}(x_j) = \begin{cases} 0, & \text{if} \quad x_j \leq 5.939 \\ \dfrac{x_j - 5.939}{6.936 - 5.939}, & \text{if} \quad 5.939 \leq x_j \leq 6.936 \\ 1, & \text{if} \quad x_j > 6.936 \end{cases}$$

$$\mu_{1,a_2}(x_j) = \begin{cases} 0, & \text{if} \quad x_j \leq 3.124 \\ \dfrac{x_j - 3.124}{3.27 - 3.124}, & \text{if} \quad 3.124 \leq x_j \leq 3.27 \\ 1, & \text{if} \quad x_j > 3.27 \end{cases}$$

These membership functions are shown in Figure 5.

To classify a new example x_0, we first determine to which cluster the new example belongs by using the obtained fuzzy membership functions. If l denotes the index of the cluster to which x_0 belongs to, we have

$$l = \operatorname{argmax}_{1 \leq i \leq k} \{\mu_i(x_0)\}.$$

Next, using the fuzzy membership function of the chosen cluster, we evaluate the memberships of the class centers $(c_1,\ c_2,\ \ldots,\ c_t)$ to determine the class.

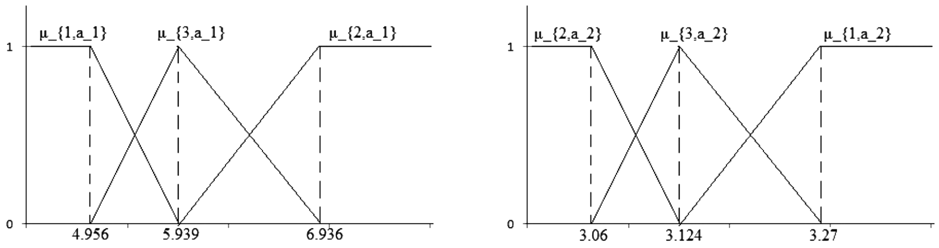


Figure 5. Membership functions for attributes a_1 and a_2.

Hence,

$$c_0 = \text{argmax}_{1 \leq j \leq t}\{\mu_l(c_j)\}$$

will be the class assigned to the new example x_0.

These steps can be summarized by

$$FBC(x_0) = \text{argmax}_{1 \leq j \leq t}\{\mu(c_j|x_0)\},$$

where

$$\mu(c_j|x_0) = \frac{\mu_l(c_j)}{\max\{\mu_1(x_0), \ldots, \mu_l(x_0), \ldots, \mu_k(x_0)\}},$$

assuming that $\max\{\mu_i(x_o)\} > 0$.

For example, let $x_0 = (5.5, 3.5)$ be a new example to be classified. We have already constructed the membership functions of each clusters in Figure 5. Therefore, when the membership function of this new example with respect to each cluster is computed it is seen that it belongs to the third cluster (see Figure 6). After determining the membership function that will be used for evaluating the membership degrees of each classes, the new example will be assigned to the class whose center has the maximum value of belongingness. It is observed that this new example belongs to the first class, which is Setosa (see Figure 7).

4. APPLICATION

In this section, we have applied Xiang et al.'s algorithm on two different data sets and compared the results for different cases of d with different must-link and cannot-link sets. The resulting data clusterings are analyzed. Moreover, the proposed method is performed on these data sets and the results are given. In Section 4.1, all examples in the Fisher Iris data set are used for the training set and the testing set. In Section 4.2, generalization performances in different classification methods are given.

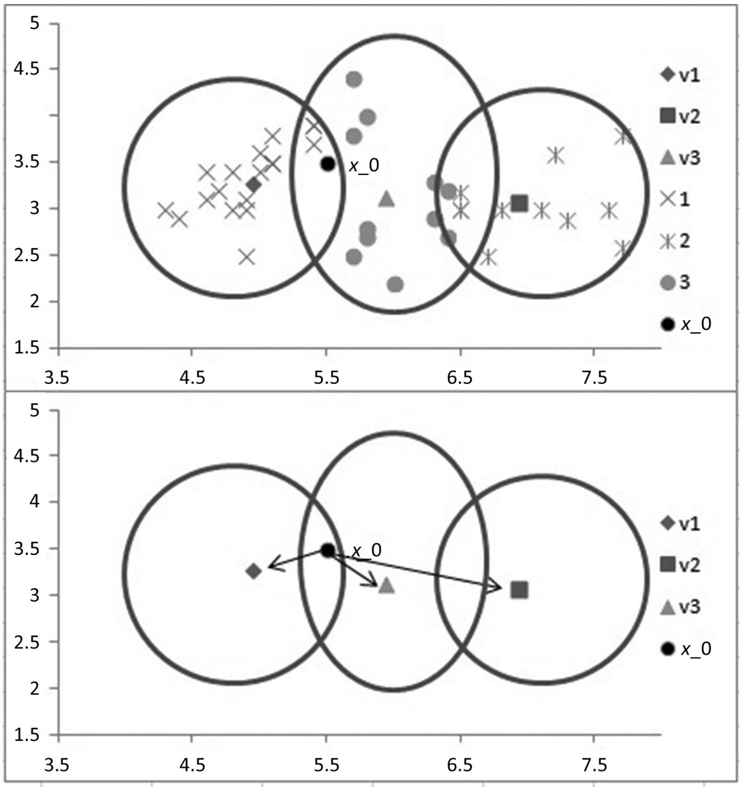


Figure 6. First step of the classification for a new example.

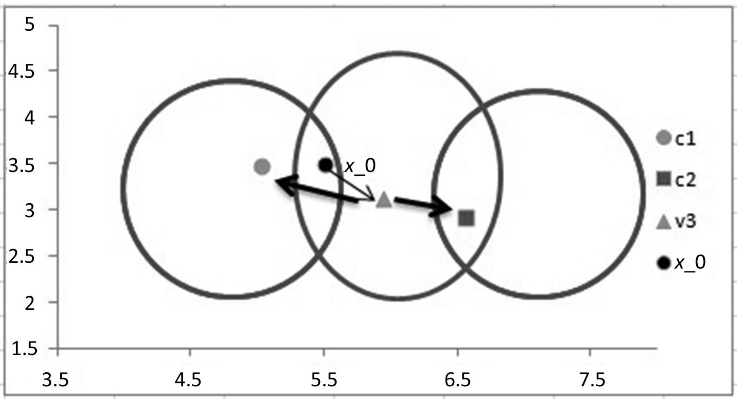


Figure 7. Second step of the classification for a new example.

Table I. Data description.

Database	Number of Data	Number of Tr	Number of Ts	Number of Attributes	Number of Classes
Iris	150	120	30	4	3
Seed	210	168	42	7	3

Table II. Training performance of Fisher Iris Data Set.

Method	Dimensionality(d)	Accuracy in clustering	Accuracy in classification
Learned Mahalanobis when			
($\|S\| = \|D\| = 20$)	1	0.960	**0.960**
	2	**0.986**	0.953
	3	0.900	0.953
Learned Mahalanobis when			
($\|S\| = \|D\| = 121$)	1	0.880	0.946
	2	0.933	0.953
	3	0.,933	**0.960**
Euclidean distance		0.980	**0.960**

Bold values show best results.

The first data set is the Fisher Iris data set, which is the best known data set in the literature. It contains three classes with 150 instances (50 in each one of three classes). In the training set, 120 instances are included and the remaining 30 examples are used for testing the proposed fuzzy Bayes classification. The second data set that we have worked with is the Seed data set containing three classes with 210 instances. In this experiment, 168 instances are used for a training set and 42 instances are used for the testing set. These two databases can be obtained from UCI Machine Learning Repository.

4.1. Training Performance for Fisher Iris Data Set

The proposed classifier is first applied to the Fisher Iris data set with different values of d used in learning a Mahalanobis distance. In addition, it is also applied to the same data set with the Euclidean distance for comparison. Fisher's Iris data set consists of 150 instances with four attributes and having three classes. As a first experiment, we have chosen the cluster number as 3. In this experiment, there are a couple of cases for d. Although we obtain minimum value for objective function used in FCM when $d = 1$ we have applied the classification procedure also for $d = 2$ and for $d = 3$ to ensure that classification accuracy is better when $d = 1$. It is also important to note that minimization of the objective function is not only sufficient for the proposed classification. Since the classification is based on cluster centers, it is very crucial to get a meaningful matrix that shows cluster centers.

Note that the accuracy in clustering of the proposed method is better than for the Euclidean distance. Besides, the performance of the classification is the same for both NBC and the proposed method (for $d = 1$ and $d = 3$).

An interesting point presented in Table II is that increasing the number of pairs in the must-link and cannot-link sets does not provide the expected improvement in the performance of the algorithm.

Table III. Generalization performances of chosen data sets.

Data set	Accuracy rate (%)	
Iris	d=1	43.33
	d=2	**100**
	d=3	93.3
	Euclidean	80
	NBC	**100**
Seeds	d=1	90.48
	d=2	**92.86**
	d=3	90.48
	d=4	90.48
	d=5	90.48
	d=6	**92.86**
	Euclidean	66.66
	NBC	**90.48**

Bold values show best results.

4.2. Generalization Performance

We have applied the proposed approach to two real data sets obtained from repository of Machine Learning data set, namely, Fisher Iris data set and Seeds database, which consist of only numerical attributes. A brief description of the data sets is given in Table I, where the number of data denotes the number of examples in the data set, the number of Tr denotes the number of training instances, and the number of Ts denotes the number of testing instances.

The proposed classifier is compared with the NBC classifier. In addition, to see the effect of distance learning the same method is applied to the Euclidean distance. The accuracy rate of classification is computed as

$$\frac{\text{Number of correctly classified instances}}{\text{Number of classified instances}} 100.$$

The results presented in Table III show that, for the Iris testing data set, the proposed method with the learned Mahalanobis distance outperforms the same method with the Euclidean distance. However, in that case performances of NBC and our classifier seem to be the same. When we look at the Seeds data set, we see that generalization performance of the proposed method is better compared to NBC.

5. CONCLUSIONS AND FURTHER STUDIES

In this study, a new classification method, which is called the fuzzy Bayesian classifier, is proposed. The proposed method is applied to Fisher's Iris data and Seed data with the Euclidean and learned Mahalanobis distances. These data sets are chosen since classes for these data sets are known. The FCM clustering algorithm is applied to achieve an optimal fuzzy partition. Based on this partition, fuzzy membership functions for each attribute are constructed, which are then used in the classification. Since in the proposed Fuzzy Bayes Classifier (FBC), there are several parameters to be considered, such as the number of clusters and the reduction parameter d, several cases were examined. The results show that changing the distance from the Euclidean distance to the Mahalanobis distance increases the classification success rate. It is also seen that, for generalization, the effect of the distance becomes more important (see Table III). As a consequence, the results for the considered data sets show that the new FBC is an effective and efficient method for the classification. The performance of the proposed FBC needs to be investigated further with respect to different parameters such as the dimension size and number of classes. We note here that a well-designed simulation study will be needed to analyze the performance of the proposed method. A further direction for research is to extend our implementation for both linguistic and numerical variables.

References

1. Pekelska E, Paclik P, Duin RPW. A generalized kernel approach to dissimilarity-based classification. Eur J Oper Res 2001;2:175–211.
2. Khemchandani R, Jayadeva, Chandra S. Learning the optimal kernel for Fisher discriminant analysis via second order cone programming. J Mach Learn Res 2010;203:692–697.
3. Li J, Lu BL. An adaptive Euclidean distance. Pattern Recog 2009;42:349–357.
4. Xiang S, Nie F, Zhang C. Learning Mahalanobis distance metric for data clustering and classification. Pattern Recog 2008;41:3600–3612.
5. Greco S, Matarazzo B, Slowinski R. Rough sets theory for multicriteria analysis. Eur J Oper Res 2001;129:1–47.
6. Duda RO, Hart PE. Pattern Classification and Scene Analysis. Wiley: New York; 1973.
7. Harry Z, Sheng S. Learning weighted naive Bayes with accurate ranking. In: Fourth IEEE Int Conf on Data Mining ICDM'04, 2004. pp 567–570.
8. Yager RR. An extension of the naive Bayesian classifier. Inform Sci 2006;176:577–588.
9. Störr HP. A compact fuzzy extension of the naive Bayes classifier based on fuzzy clustering. IEEE Int Conf on Syst Man Cybernet 2002;176:1–6.
10. Tang Y, Pan W, Li H, Xu Y. Fuzzy naive Bayes extension of the naive Bayes classifier based on fuzzy clustering. IEEE Int Conf Syste Man Cybernet 2002;56:1–6.
11. Chen CB, Wang LY. Rough set-based clustering with refinement using Shannon's entropy theory. Comput Math Appl 2006;56:1563–1576.