IACSIT International Journal of Computer Theory and Engineering, Vol. 1, No. 5, December 2009
ISSN: 1793-8236

Fuzzy Bayesian Classification of LR Fuzzy Numbers

Hadi Sadoghi Yazdi, Mehri Sadoghi Yazdi , Abedin Vahedian

Abstract— **Fuzzy data is considered as an imprecise type of data with a source of uncertainty. Fuzzy numbers allow us to model uncertainty of data in an easy way which justifies the increasing interest on theoretical and practical aspects of fuzzy arithmetic. This paper presents a Fuzzy Bayesian Classifier (FBC) over LR-type fuzzy numbers with unknown conditional probability density function. A new version of K-NN method is used to estimate conditional probability density function for Bayesian classification of fuzzy numbers.**

Fairly good recognition rate has been obtained over fuzzy numbers in classification using FBC even in the presence of noise.

Index Terms— **fuzzy data, Bayesian classifier, LR-type fuzzy numbers.**

I. Introduction

In the field of statistical pattern recognition, Bayesian decision theory is a basic approach. In many practical situations, obtaining density function is fairly difficult. The density function is assumed to have a particular type of distribution. Complicated testing and advanced statistical techniques are employed to estimate the parameters of the selected distribution.

In the statistical learning theory, random samples are considered for obtaining statistical information such as mean and variance. In the real world, however, we always encounter fuzzy information in which probability density functions (PDF) parameters are assumed to form fuzzy numbers, as in [1, 2]. In this group PDF of data has been assumed to have the form of:

$$g(x) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left[-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2\right]$$

Where μ is a fuzzy number which in [2] is assumed to be a triangular fuzzy number. In the above works, x is a crisp random variable and only its mean value is considered as a fuzzy number. In [3] a method for probability density function approximation is presented using fuzzy rules for encoding a density function of a crisp random variable. As it can be seen, none of the mentioned works use fuzzy random variables. In this paper we introduce density estimation of fuzzy number based on a new type of k nearest neighbor algorithm. We begin with reviewing some approaches about fuzzy Bayesian classification in the literature before using the estimated density function for Bayesian classification.

In [4, 5] a fuzzy Bayesian approach provide an alternative technique to obtain the conditional density function without the assumption on the type of distribution. The likelihood density function is estimated based on the likelihood conditional probability with fuzzy supported values. In order to estimate likelihood conditional density function it uses the equation (1):

$$f^*\left(e\middle|H_j\right) = \frac{\sum_{i=1}^{n} W(e_i)}{U}\left(\begin{array}{l}\frac{\mu_{e_1}(e)}{W(e_1)}P\left(e_1\middle|H_j\right)+ \\ \frac{\mu_{e_2}(e)}{W(e_2)}P\left(e_2\middle|H_j\right)+\ldots \\ +\frac{\mu_{e_n}(x_0)}{W(e_n)}P\left(e_n\middle|H_j\right)\end{array}\right) \quad (1)$$

Where $W(e_i)$ refers to the size of interval of e_i and U is the size of universe of discourse. The likelihood conditional probabilities with fuzzy values $P\left(e_i\middle|H_j\right)$ are observed from professional experiences. Likelihood density function is computed as the weighted sum of the likelihoods for all of fuzzy sets. The weight is proportional to the corresponding degree of belonging obtained from the membership functions and is inversely proportional to the corresponding size of the fuzzy set interval. It also imposes a restriction on the membership functions so that for any particular value e in the universe of discourse, the sum of $\mu e_i(e)$ is equal to 1. For a given conditional probability of the fuzzy values and the size of the fuzzy set intervals, the shape of estimated conditional density function depends on the shape of the membership functions; if more fuzzy sets of the evidence are available, the estimated conditional density function is more accurate.

In the other work, [6] proposed a fuzzy rule-based classifier with Bayesian rule that yields to prune the irrelevant features consequently. It has also assumed that discriminant function is a Gaussian membership function and has proposed some assumptions for avoiding the weak points of Bayesian classifier like the singularity of covariance matrix and the difficulty of feature selection. It, however, does not use fuzzy

Hadi Sadoghi Yazdi is with the Computer Engineering Department, Ferdowsi University of Mashhad, IRAN

Mehri Sadoghi Yazdi is master student in Electrical and Computer Engineering Department, Shahid Beheshti University; G.C Tehran, Iran

Abedin Vahedian is with the Computer Engineering Department, Ferdowsi University of Mashhad, IRAN

data.

In [7] a generalized Naive Bayesian classifier is proposed that uses the fuzzy partition of variables instead of discretizing them. It partitions the domain of each continuous variable into fuzzy regions. Therefore, each variable is a linguistic variable taking linguistic values. The training of Fuzzy Naive Bayesian classifier is done by performing an unsupervised fuzzy clustering in the feature space to obtain an optimal fuzzy partition. The conditional probabilities of each node in Fuzzy Naive Bayesian classifier are then estimated.

In the current work, though, we present Bayesian classifier over fuzzy numbers as a quite new approach. In similar works with the proposed method [4 and 5], density estimation over fuzzy data (both discrete and continuous) have been studied with known density function according to following equation:

$$P(e_i|H_j) = \int_{x\in e_I} \mu_{e_i}(x)\, f(x|H_j)\, dx \qquad (2)$$

Where $f(x \mid H_j)$ is the conditional probability density function at value x given H_j. But it is assumed that $f(x \mid H_j)$ is known. This assumption in practical cases in real world is not possible. In the present work, we try to find PDF of fuzzy numbers using the proposed fuzzy K nearest neighbor (K-NN) algorithm that is a well known artificial intelligence algorithm. Dominant notes of the proposed Fuzzy Bayesian Classifier (FBC) are,

- Using LR fuzzy numbers in Bayesian Classifier
- Generation of probability density function using K Nearest Neighbor (K-NN) algorithm
- Flexibility of the proposed classifier over noisy data
- Probability density estimation using various fuzzy distance metrics.

The reminder of this paper is organized as follows: In section 2 some preliminaries about fuzzy numbers, KNN algorithm and Bayesian classification are reviewed. Our proposed approach for fuzzy density estimation using KNN method together with fuzzy Bayesian classification over LR-type fuzzy numbers is discussed in section 3. Section 4 represents experimental results of the proposed method. Finally, section 5 represents a conclusion of the paper.

II. Preliminaries

Some definitions about fuzzy numbers, K-nearest neighbor algorithm and Bayesian classification are explained briefly.

A. Definition of fuzzy numbers

Consider $\tilde{A}$ as a fuzzy number, then Alpha-cut of $\tilde{A}$ is shown by $\tilde{A}_\alpha = \{x : \mu_{\tilde{A}} \geq \alpha\}$ which is a closed interval and is denoted to $\tilde{A}_\alpha = [A_\alpha^L, A_\alpha^U]$, where $\alpha \in [0,1]$.

In order to compute the distance of two fuzzy numbers, several formulas are proposed. One is Hausdorff distance described below:

For any two fuzzy numbers $\tilde{A}, \tilde{B} \in F$, Hausdorff distance metric is defined by [8]:

$$d_F[\tilde{A},\tilde{B}] = \max\left\{\left|\tilde{A}_\alpha^L - \tilde{B}_\alpha^L\right|, \left|\tilde{A}_\alpha^U - \tilde{B}_\alpha^U\right|\right\} \qquad (3)$$

Another metric is Hathaway distance. Let $\tilde{A} = (a_1, a_2, a_3, a_4)$ and $\tilde{B} = (b_1, b_2, b_3, b_4)$ are two fuzzy data, where we refer to a_1 as the center, a_2 as the inner diameter, a_3 as the left outer radius and a_4 as the right outer radius. Then Hathaway distance $d_h(\tilde{A},\tilde{B})$ is defined as [9]:

$$d_h(\tilde{A},\tilde{B}) = (a_1 - b_1)^2 + (a_2 - b_2)^2 + (a_3 - b_3)^2 + (a_4 - b_4)^2 \qquad (4)$$

Yang distance is another metric [10] which is defined for all LR-type fuzzy numbers by considering a fuzzy number $\tilde{X}$ with its membership function $\mu_{\tilde{X}}(x)$:

$$\mu_{\tilde{X}}(x) = \begin{cases} L\left(\dfrac{m_1 - x}{\alpha}\right) & for\, x \leq m_1 \\ 1 & for\, m_1 \leq x \leq m_2 \\ R\left(\dfrac{x - m_2}{\beta}\right) & for\, x \geq m_2 \end{cases} \qquad (5)$$

This is called an LR-type TFN (Trapezoidal Fuzzy Number). In Fig. 1, two types of TFN are shown.

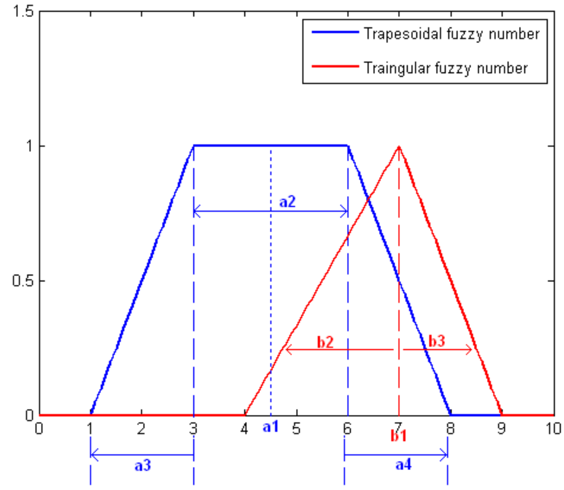


Fig. 1 Two samples of TFN

For any two fuzzy numbers $\tilde{A}, \tilde{B}$ that are shown as fuzzy numbers in Hathaway distance, the distance $d_f(\tilde{A},\tilde{B})$ based on Yang definition is:

$$d_f^2(\tilde{A},\tilde{B}) = \frac{1}{4}\begin{pmatrix} g_-^2 + g_+^2 + (g_- - (a_3 - b_3))^2 \\ + (g_+ - (a_4 - b_4))^2 \end{pmatrix} \qquad (6)$$

Where $g_- = 2(a_1 - b_1) - (a_2 - b_2)$ and $g_+ = 2(a_1 - b_1) + (a_2 - b_2)$.

B. K nearest neighbor algorithm

There are several methods to estimate an unknown probability density function. The most fundamental techniques to estimate $p(x)$ use the following procedure:

Firstly, a sequence of regions $R_1, R_2, \ldots$ containing x are formed so that the first region is used with one sample, the second with two, and so on. Also let V_n be a volume of R_n, k_n be the number of samples falling in R_n, and $p_n(x)$ be the nth estimate for $p_n(x)$, then:

$$p_n(x) \approx \frac{k_n/n}{V_n} \tag{7}$$

If the three conditions below are satisfied then $p_n(x)$ converge to $p(x)$:

- $\lim_{n\to\infty} V_n = 0$
- $\lim_{n\to\infty} k_n = \infty$
- $\lim_{n\to\infty} k_n/n = 0$

With these basics for K-nearest neighbors algorithm, in order to estimate $p_n(x)$ from n training samples or *prototypes* one can center a cell about **x** and let it grow until its prototype captures k_n samples. These samples are the k_n nearest neighbors of **x**. If the density is high around **x**; the cell then will be relatively small, which leads to a good resolution. If the density is low, however, the cell will grow fast, but will stop soon after it enters regions of higher density. Considering equation (7) and its conditions assures that the ratio k_n/n is a good estimation of the probability that a point fall in the cell of volume V_n [11].

C. Bayesian classification

For classification of pattern x into classes w_j we use the conditional probability $p(w_j|x)$. Suppose that both the prior probabilities $P(W_j)$ and the conditional densities $p(x|w_j)$ are known. Bayes definition is then stated in equation (8):

$$p(w_j|x) = \frac{p(x|w_j)P(w_j)}{p(x)} \tag{8}$$

If the number of classes is N, $p(x)$ is then obtained according to [5] as indicated in equation (9):

$$p(x) = \sum_{i=1}^{N} p(x|w_j)P(w_j) \tag{9}$$

$p(w_j|x)$ is posterior probability and is computed using likelihood probabilities $p(x|w_j)$ and prior probability $P(W_j)$. A Bayesian classifier employs these probabilities and then using a decision rule, it classifies samples into related classes. Decision rule is defined as follows:

$$\begin{array}{ll} \text{if} \quad p(w_i|x) > p(w_j|x) \text{ then } x \in w_i \\ \text{else} \quad\quad\quad\quad\quad\quad\quad\quad x \in w_j \end{array} \tag{10}$$

So for state of $x \in w_i$ we have,

$$\frac{p(x|w_i)P(w_i)}{p(x)} > \frac{p(x|w_j)P(w_j)}{p(x)} \tag{11}$$

As it can be seen in Equation (8), likelihood probability must be known. In the practical cases, however, this density function is not known and it must be estimated. As it will be seen in more detail in the next section, we use K-NN approach to estimate this density function.

In the next section, we develop Bayesian classifier to FBC. Practically, data samples are noisy and uncertainty is assumed because of outliers in samples. In Bayesian classification of fuzzy numbers, calculation of $p(w_i|\tilde{x})$ with unknown $f(x \mid H_j)$ is an open problem in fuzzy mathematics. Fuzzy number classification will be seen from viewpoint of K-NN for calculation of $f(x \mid H_j)$.

III. The Proposed Method

The proposed method is based on a Fuzzy Bayesian classifier over LR-type fuzzy numbers. Fuzzy Bayes formula is introduced with the following equation:

$$p(w_j|\tilde{x}) = \frac{p(\tilde{x}|w_j)P(w_j)}{p(\tilde{x})} \tag{12}$$

In order to use Bayesian classification for fuzzy numbers, it is required to compute $p(\tilde{x}|w_i)P(w_i)$ for each class w_i. In N-dimensional feature space, samples are in the form of $\tilde{x} = (\tilde{x}_1, \tilde{x}_2, \ldots, \tilde{x}_n)$. For instance, in 2-D feature space, we have pairs of LR-fuzzy numbers such as $(\tilde{2}, \tilde{1})$. Fig. **2** depicts some samples.

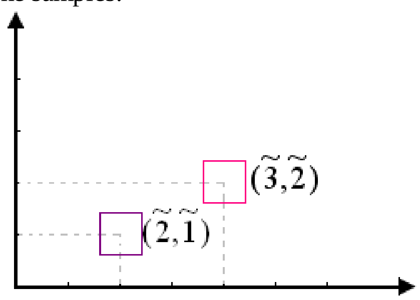


Fig. 2 Two fuzzy numbers in 2-D space

As the density function $p(\tilde{x}|w_i)$ is unknown, it must be then estimated. We use new version of K-NN method to estimate this density function in this work.

A. Fuzzy number density estimation by new version of K-NN

In the previous section some details about K-NN method were presented stating that its formulation is over crisp numbers. In this section, we present a new method for estimating probability density function of a fuzzy variable by using of K-NN algorithm. We start by formulations of K-NN over fuzzy numbers considered as follows:

$$p_n(\tilde{x}) \approx \frac{\tilde{k}_n / n}{\tilde{V}_n} \tag{13}$$

In the above equation $\tilde{x}$ is a fuzzy variable, $\tilde{k}_n$ are the nearest neighbors samples of the $\tilde{x}$.

In the proposed algorithm several fuzzy numbers are created, each of which has a different label. Using K-NN method, then, we find nearest neighbors of each sample by a distance metric like those metrics defined in the previous section. Finally the number of each label is counted and its histogram is plotted. As we can see in the next section, this histogram is reliable density estimation for fuzzy data. Using more and more nearest neighbors or alternatively decreasing the radius of neighborhood, the density estimation obtains better results similar to the discussions about K-NN method in section 2.

The following steps describe the presented method:

KNN Based Density Estimation:

Step1: Generate some fuzzy numbers with arbitrary distribution (e.g. triangular fuzzy numbers with one or two Gaussian probability density function, two samples are shown in Fig. 3).

Step2: Assign a different label to each generated fuzzy number for characterizing each fuzzy number as an index which are used in the next steps. For example, assign label L_1 to the first sample, L_2 to the second sample and so on.

Step3: Apply K-NN algorithm in order to find nearest neighbors of each test sample by using a distance metric (e.g. Hausdorff, Hathaway and Yang distance). Assume that training and testing samples for K-NN algorithm are the same and equal to the created fuzzy set in step1, as we want to estimate density functions of the created fuzzy numbers, such that the training and test samples in K-NN algorithm be equal. In order to obtain nearest neighbor samples for each test sample we should compute its distance with all of the training samples and find labels of samples that their distance with test sample is below a given threshold (this threshold is the radius of neighborhood).We must keep these labels for each test sample. For instance, if one of the created triangular fuzzy numbers with one Gaussian PDF is $[-0.4758 - 0.4326 - 0.3893]$ assumed as a test sample, calculated nearest neighbors for this test sample with radius 0.99 using Yang distance include 294 training sample like 1, 8, 11, 13, …, 799. Some of these fuzzy numbers are shown in Fig. 3.

Step4: After we obtain all of the labels of nearest neighbors for each sample we generate in the first step, we compute histogram of these labels which are the indices of the nearest neighbors of each test sample. This histogram, indeed, represents the frequency of the occurrence of each label in all sets of nearest neighbors per sample. This histogram shows estimated PDF of generated fuzzy numbers.

Step5: Finally, we compute estimation error using equation (14).

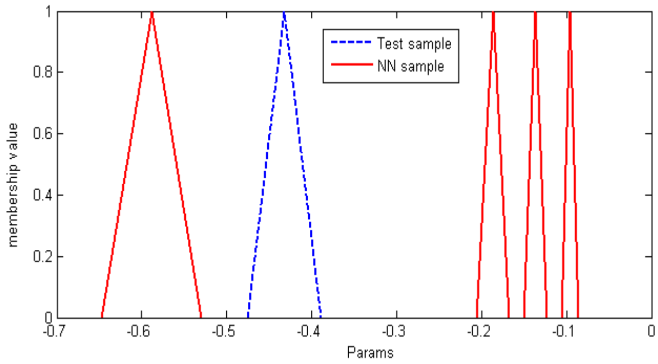


Fig. 3 A number of computed nearest neighbors for determined test sample

Distribution of the nearest neighbors of fuzzy numbers (with any arbitrary density function) can be encountered as an estimator of its probability density function. In the next section we show this result with some experiments. The proposed method is robust in the presence of different amount of noise as well.

B. Fuzzy Bayesian classifier

Now fuzzy Bayesian classifier is applied to assign each test sample to the correct class and recognition rate is calculated. The following steps show our proposed approach for 2 class Bayesian classifier:

Fuzzy Bayesian Classification:

Step1: Create two classes of LR-type fuzzy numbers with arbitrary distribution.

Step2: Consider test samples from one of the created classes.

Step3: Apply K-NN algorithm in order to estimate likelihood density function by using a distance metric. (e.g. Hausdorff, Hathaway and Yang distance).

Step4: Using the estimated likelihood density function, compute a confusion matrix that includes the probability of belonging test samples into classes.

Step5: Finally, the recognition rate is achieved from obtained confusion matrix.

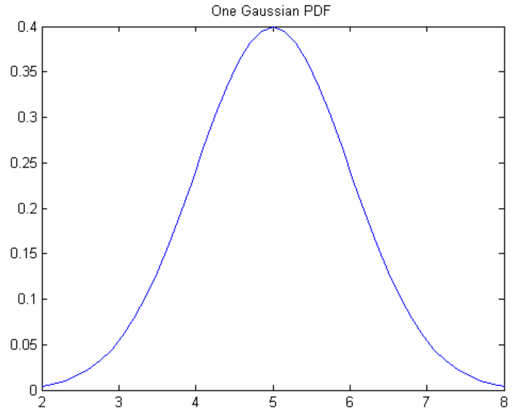
IV. Experimental Results

The proposed method has been implemented in KNN-based density estimation and then fuzzy Bayesian classifier.

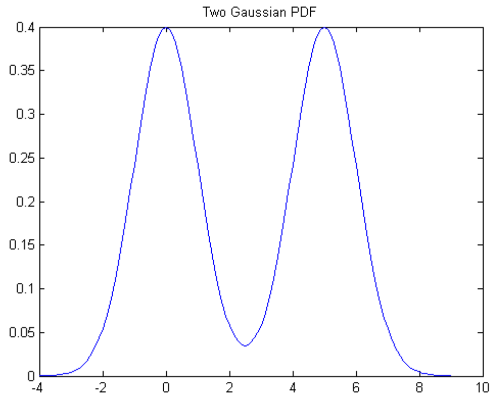
A. Density estimation using KNN algorithm

We have implemented density estimation using KNN method for triangular fuzzy numbers. Different distance metrics are used such as Hausdorff, Hathaway and Yang. We created 400 triangular symmetric fuzzy numbers which belong to two classes of fuzzy numbers. One is zero mean, one Gaussian density function, with a variance of 1 and the other has two Gaussian density functions with mean values as 0 and

5 and a variance of 1. Fig. 4 shows a typical one and two Gaussian PDF with former description.



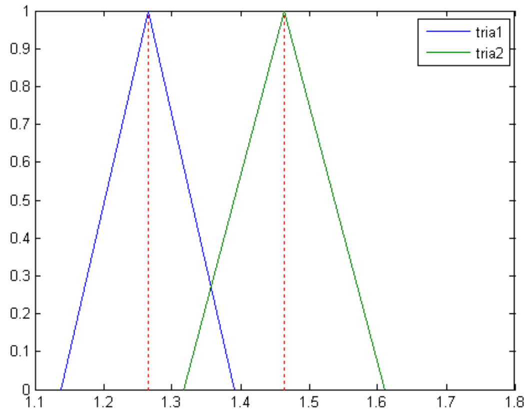
(a)



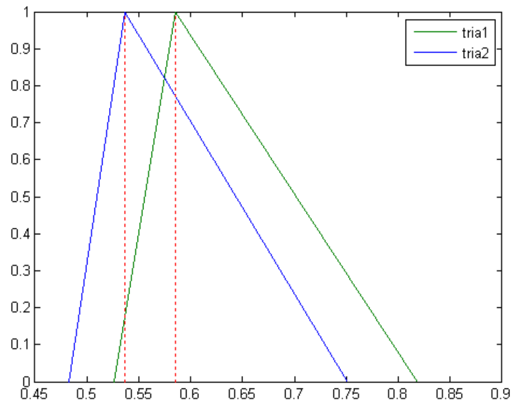
(b)

Fig. 4 Typical (a) one and (b) two Gaussian PDF

Fig. 5 indicates samples of triangular symmetric and asymmetric fuzzy numbers:



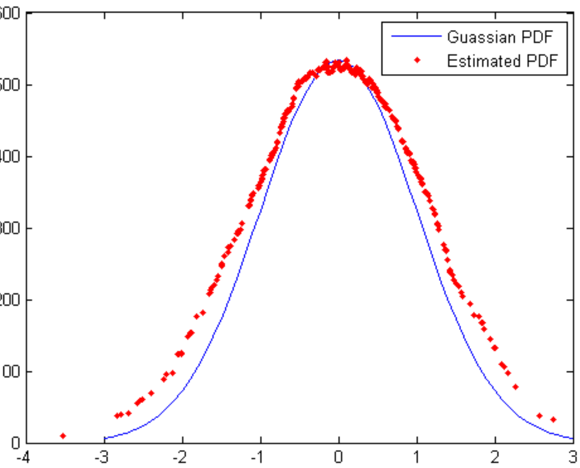
(a)



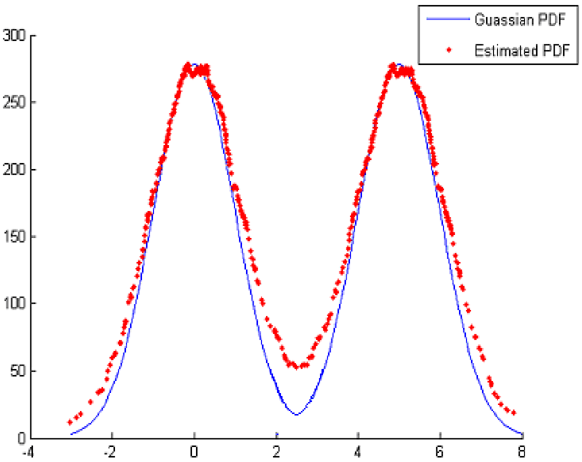
(b)

Fig. 5 Triangular (a) symmetric and (b) asymmetric fuzzy data

We then ran our method over these numbers. Fig. 6 shows the estimated PDF with Hathaway distance and radius of 0.99 compared to a normal one and two Gaussian PDF:



(a)

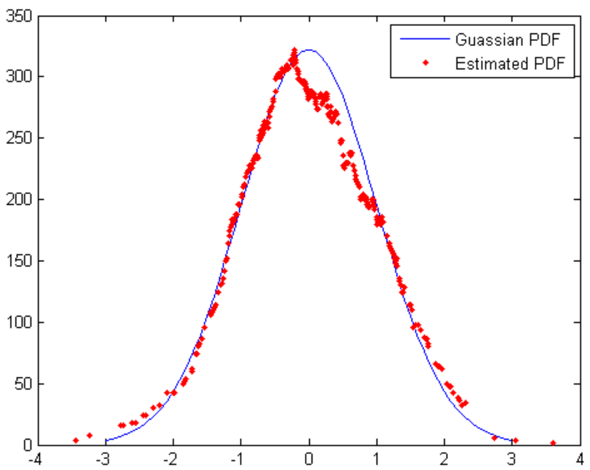


(b)

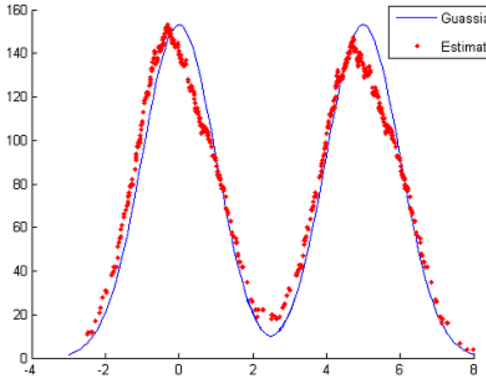
Fig. 6 Density estimation of (a) one and (b) two Gaussian PDF of fuzzy numbers with Hathaway's distance metric

It is important to note that the PDF of non-symmetric fuzzy numbers is obtained as well.

Figure 7 represents the result of estimated PDF with Yang distance for fuzzy triangular data that are not symmetric.



(a)



(b)

Fig. 7 Density estimation of (a) one and (b) two Gaussian PDF of asymmetric fuzzy numbers with Yang distance metric

We now consider our method for LR-type fuzzy numbers described in equation (13). A set of 400 LR-type fuzzy numbers are created with one and two Gaussian distribution, then we ran our method on these data. The result of density estimation appears in Fig. 8 as two LR-type triangular fuzzy numbers.

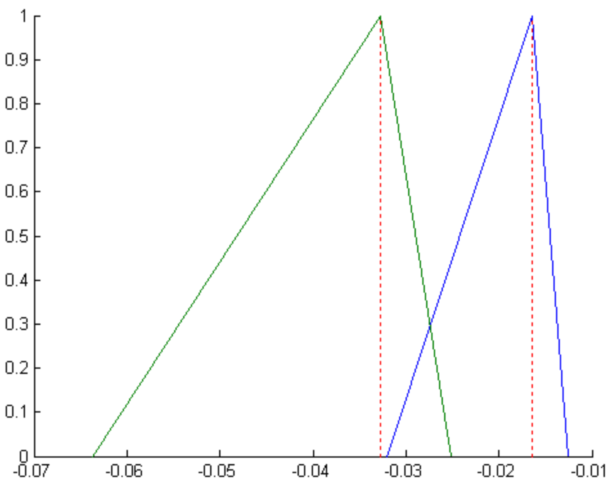
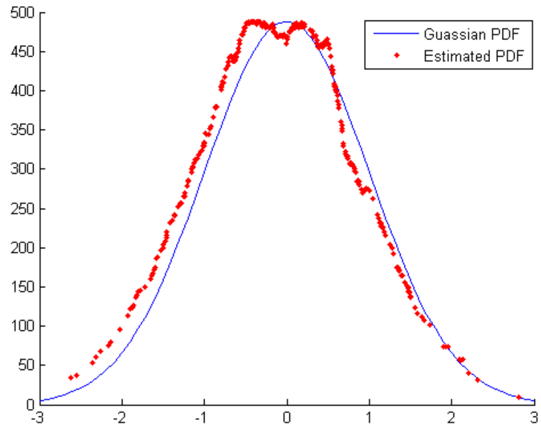
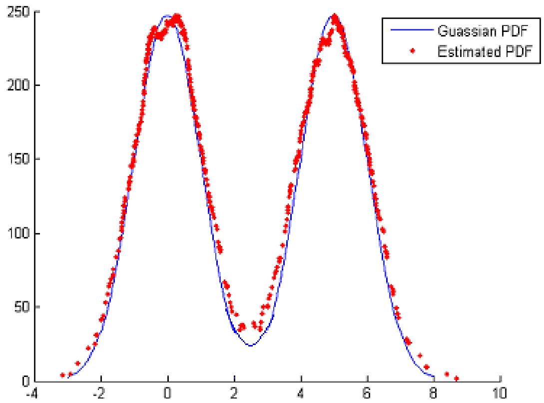


Fig. 8 LR-type triangular fuzzy numbers

With different distance metrics, the results of PDF estimation of LR-type fuzzy data with Hausdorff distance metric is shown in figure 9.



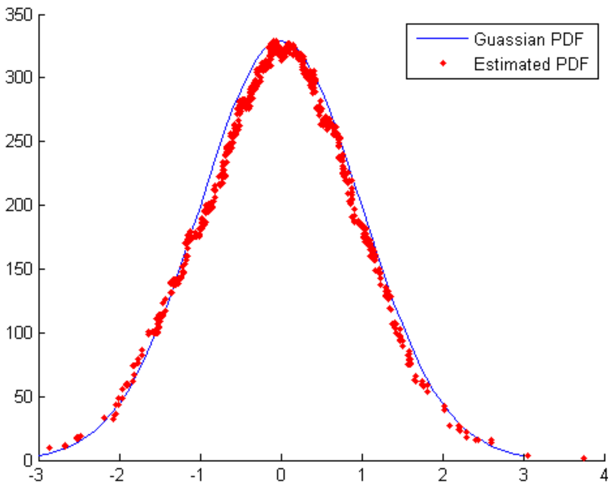
(a)



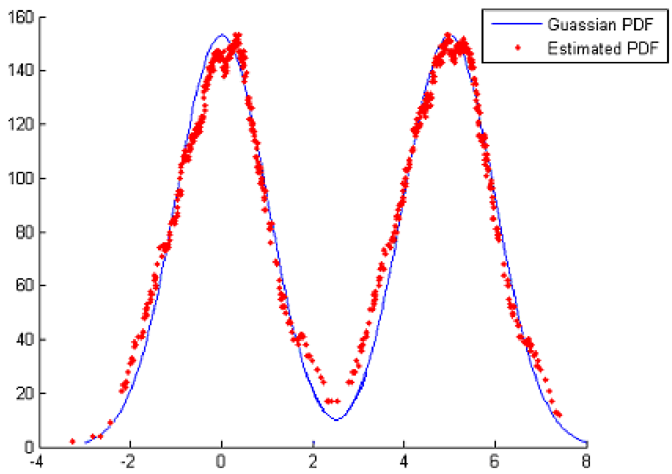
(b)

Fig. 9 Density estimation of (a) one and (b) two Gaussian PDF of LR-type fuzzy numbers with Hausdorff distance metric

This method can also estimate PDF of noisy data. A 10% noise has been added to the training fuzzy data, Fig. 10 represents the result of density estimation with Yang distance metric.



(a)



(b)

Fig. 10 Probability density estimation of (a) one and (b) two Gaussian PDF LR-type noisy fuzzy numbers with Yang distance metric

At this stage, we compare accuracy of PDF estimation by different distance metrics. In order to obtain the average estimation error of each distance metric, we ran our algorithm 100 times both over one and two Gaussian fuzzy data and use the following equation for computing estimation error:

$$Average\ Error = \frac{1}{N}\sum\left|E.D.F - O.D.F\right| \tag{14}$$

Where E.D.F is abbreviate form of estimated density function, O.D.F is abbreviation of original density function and N is the number of times we run the algorithm, it was set to be 100. Estimated density function is proportionate to given data and computation defined error. We, however, need the difference of estimated and original density function values that we considered in this experiment. Table I summarizes the results.

Table I Average error for each distance metric

Distance Metric	Average error (for one Gaussian fuzzy data)	Average error (for two Gaussian fuzzy data)
Hausdorff distance	270.5028	74.2085
Hathaway distance	270.5028	74.2085
Yang distance	148.1467	38.2023

From the above table one can conclude that Yang distance has less average error value than the other metrics. Fig. 11 shows a plot of average error values for density estimation of two Gaussian PDFs over 100 run for different distance metrics.

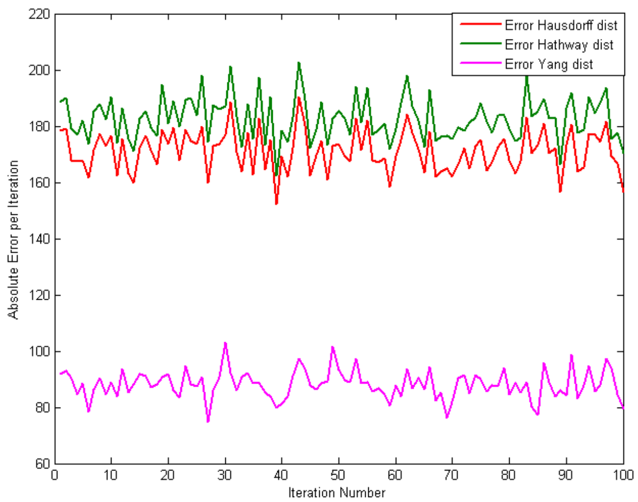


Fig. 11 Average error of different distance metrics for estimation of a two Gaussian PDF

B. *Fuzzy Bayesian Classification*

In this section we discuss the results of implementing the proposed method on fuzzy Bayesian Classifier. The method was performed on LR-type fuzzy numbers, with one and two Gaussian distributions as shown in Fig. 4. Fig. **12** displays some LR fuzzy samples of generated training classes.

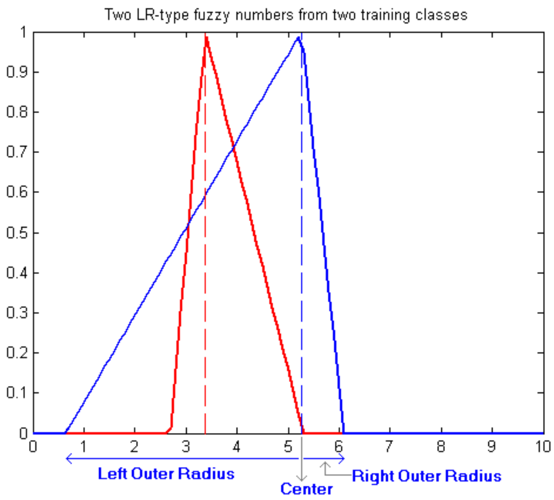


Fig. 12 LR-type fuzzy numbers from two classes

Table II also represents the average recognition rate over 100 runs of the proposed algorithm for three different distance metrics i.e. Hausdorff, Hathaway and Yang distance.

Table III. Average recognition rate for each distance metric

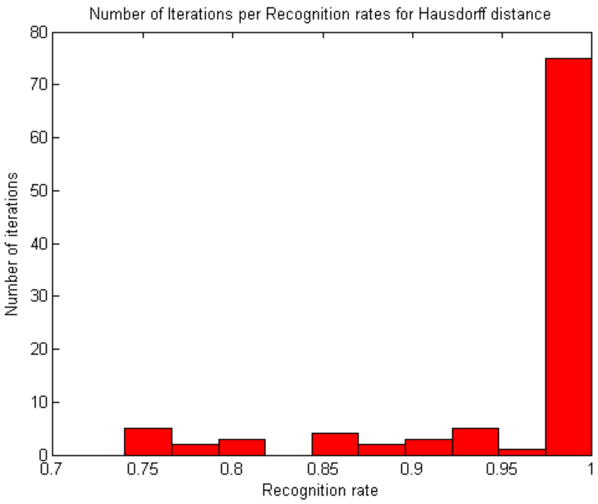
Distance metric	Mean recognition rate
Hausdorff	0.9605
Hathaway	0.9781
Yang	0.9153

Confusion matrixes with respect to three distance metrics are averaged over 100 runs. Table III indicates the results.

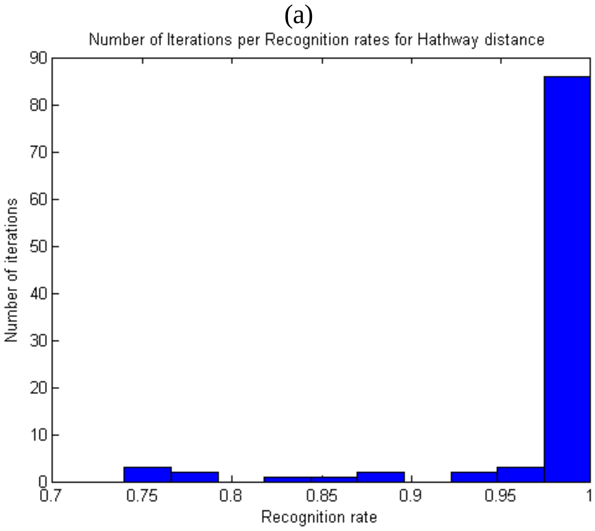
Table III. Average Confusion Matrix for each distance metric

Distance metric	Confusion Matrix
Hausdorff	$\begin{bmatrix} 0.9265 & 0.0054 \\ 0.0735 & 0.9946 \end{bmatrix}$
Hathaway	$\begin{bmatrix} 0.9593 & 0.0030 \\ 0.0407 & 0.9970 \end{bmatrix}$
Yang	$\begin{bmatrix} 0.8378 & 0.0072 \\ 0.1622 & 0.9928 \end{bmatrix}$

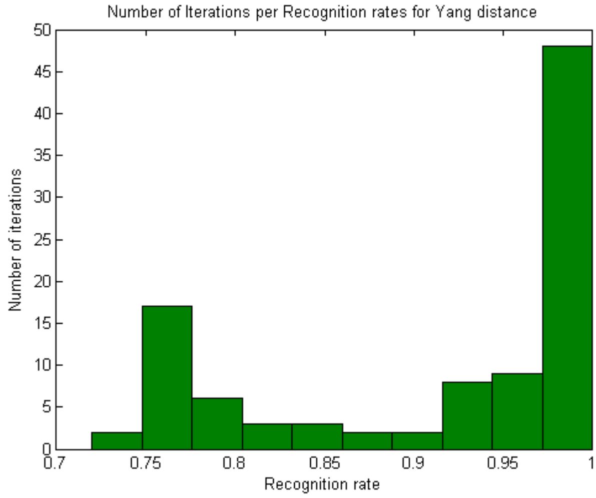
As it can be seen in Table II, Hathaway distance makes highest recognition rate and Yang distance has the most misclassification. Figure 13 shows the histogram of recognition rates of 100 run of the proposed method for different distance metrics.



(a)



(b)



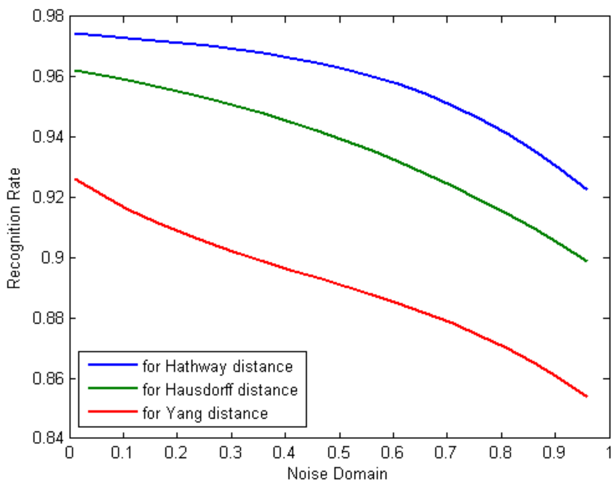
(c)

Fig. 13 Number of Iterations per Recognition rates for (a) Hausdorff (b) Hathaway and (c) Yang distance

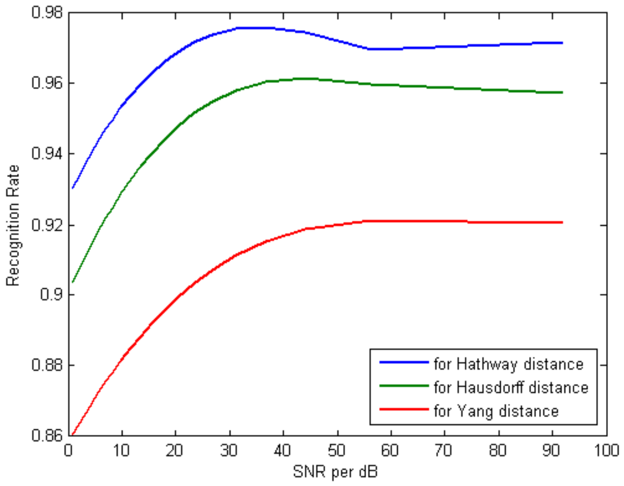
Therefore, according to Figure 13, we find that the average recognition rate of Bayesian classifier using Hathaway distance for density estimation has accurate recognition in more iteration. For instance, the number of iterations that have recognition rate equal to 1 (the last bar shown in histograms), for Hathaway distance is greater than the other metrics.

We now discuss about robustness of our proposed method against noise. Experiments are performed in the presence of noise both in the center of each training fuzzy number and in their left and right outer radius.

We have also added various values of noise into centers of each training fuzzy numbers. Fig. **14** displays recognition rate versus noise domain and SNR (Signal to Noise Ratio) for different distance metrics introduced in section 2.



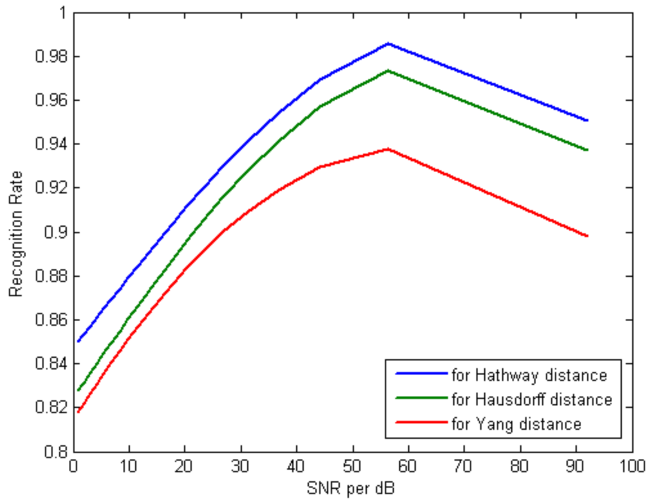
(a)



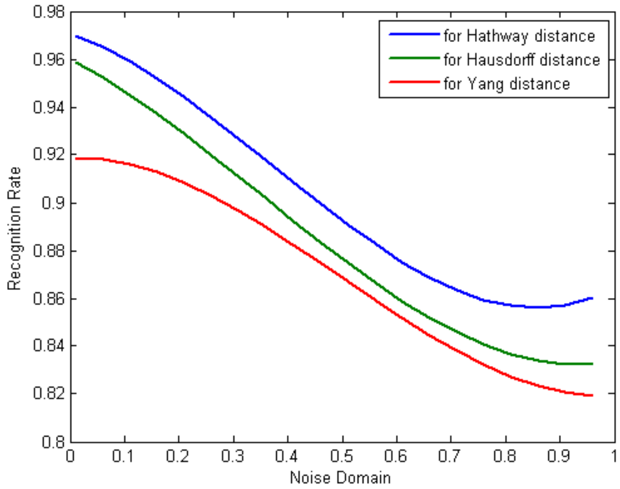
(b)

Fig. 14 Recognition rate versus (a) noise variation and (b) SNR

Various values of noise were added at this stage into left and right outer radius of each training fuzzy data. Recognition rate versus noise variations and SNR are plotted in Fig. 15.



(a)



(b)

Fig. 15 Recognition rate versus (a) noise variations and (b) SNR

V. CONCLUSION

In this paper we have proposed a Fuzzy Bayesian Classifier for LR-type fuzzy numbers with unknown density function. In order to estimate likelihood density function, a new version of KNN method was used. Experimental results indicate good estimation accuracy on KNN-based PDF estimation and a good recognition rate on Bayesian classifier even in the presence of noise variations.

REFERENCES

[1] James J. Buckley, "Fuzzy statistics: hypothesis testing", soft Computing, Volume 9, Issue 7, pp: 512 – 518, 2005.

[2] Ru-Jen Chao and Bilal M. Ayyub, "Distributions with Fuzziness and Randomness", IEEE, Proceedings of ISUMA-NAFIPS, pp.668-673, 1995.

[3] Victor Ayala-Ramirez, Raul E. Sanchez-Yanez, Oscar G. Ibarra-Manzano, Francisco J. Montecillo-Puente, "Probability Density Function Approximation using Fuzzy Rules", IEEE, Multi conference on electronics photonics , pp: 220-222, 2006.

[4] Christopher C. Yang and Ken M. Cheung "Fuzzy Bayesian Analysis with continuous-valued evidence", 0-7803-2559-1/95 1995 IEEE.

[5] Christopher C. Yang, "Fuzzy Bayesian Inference", 0-7803-4053- 1/97/ 1997 IEEE, pp: 2707-2712.

[6] Do Wan Kim, Jin Bae Park, and Young Hoon Joo, "Fuzzy Classifier with Bayes Rule Consequent", LNAI 3809, pp. 1130–1133, 2005.

[7] Yongchuan Tang, Wuming Pan, Haiming Li and Yang Xu, "Fuzzy Naïve Bayes classifier based on fuzzy clustering", IEEE SMC 2002.

[8] Jing TIAN, Ming-Hu Ha, Jun-Hua Li, Da-Zeng Tian, "The Fuzzy-Number Based Key Theorem of Statistical Learning Theory", IEEE, Proceedings of the Fifth International Conference on Machine Learning and Cybernetics, pp: 3475-3479, 2006.

[9] R.J. Hathaway, J.C. Bezdek, W. Pedrycz, "A parametric model for fusing heterogeneous fuzzy data", IEEE Trans. Fuzzy Systems, pp: 270–281, 1996.

[10] M.S. Yang, C.H. Ko, "On a class of fuzzy c-numbers clustering procedures for fuzzy data", Fuzzy Sets and Systems 84 pp: 49–60, 1996.

[11] Richard O. Duda, Peter E. Hart and David G. Stork, Pattern Classification (2nd edition), chapter 4, pp: 3-18, 1997.