DATA MINING METHODOLOGIES WITH UNCERTAIN DATA

by

BEHNAM TAVAKKOL

A dissertation submitted to the

School of Graduate Studies

Rutgers, The State University of New Jersey

In partial fulfillment of the requirements

For the degree of

Doctor of Philosophy

Graduate Program in Industrial and Systems Engineering

Written under the direction of

Professors: Susan L. Albin and Myong K. Jeong

And approved by

_____

_____

_____

_____

New Brunswick, New Jersey

May, 2018

ABSTRACT OF THE DISSERTATION

Data Mining Methodologies with Uncertain Data

By BEHNAM TAVAKKOL

Dissertation Directors:

Dr. Susan L. Albin and Dr. Myong K. Jeong

Uncertain objects arise in many applications such as sensor networks, moving object databases and medical and biological databases where each feature is represented by multiple observations or a given or fitted probability density function (PDF). In this dissertation we present a methodology to classify uncertain objects based on a probabilistic distance measure between an uncertain object and a group of uncertain objects. We call this newly proposed measure object-to-group probabilistic distance measure, OGPDM, noting that dozens of probabilistic distance measures (PDM) for the distance between two pdfs exist in the literature. To assess the accuracy of the OGPDM, we compare it to some existing classifiers, i.e., K-Nearest Neighbor (KNN) classifier on object means (certain

KNN) and uncertain naïve Bayesian classifier. In addition we compare OGPDM to an uncertain K-Nearest Neighbor (KNN) classifier, which we propose here, that uses existing PDMs to measure object-to-object distances and then classifies using KNN. We illustrate the advantages of the proposed OGPDM classifier with both simulated and real data. OGPDM captures the correlation among features within a class. Also, it takes into account the correlation among features within objects which is not taken into account in most of other uncertain data classification approaches.

Because of existing levels of uncertainty for uncertain data objects, the scatter of this type of objects might be very different than the scatter of certain data objects. Measures of scatter for uncertain objects have not been defined before. Here in this dissertation, we define measures of scatter such as covariance matrix, within scatter matrix, and between scatter matrix, for uncertain data objects. Also, we extend the idea of Fisher Linear Discriminant Analysis (LDA) for uncertain objects. We also develop Kernel Fisher Discriminant for uncertain objects. The developed Uncertain Fisher LDA produces linear decision boundaries for separating classes of uncertain data objects while the developed Uncertain Kernel Fisher Discriminants produce nonlinear decision boundaries. The developed Uncertain Kernel Fisher Discriminants are for two cases: when the uncertain objects are given with PDF, and when the uncertain objects are given with multiple points. We show through examples that the obtained decision boundaries from our developed uncertain Fisher Discriminants seem very reasonable for separating classes of uncertain objects. Also, we compare the classification performance with many existing classifiers on simulated scenarios with uncertain objects modeled with skew-normal distribution and a real-world data set.

To evaluate the quality of formed clusters and determine the correct number of clusters, clustering validity indices can be used. They can be applied on the results of clustering algorithms to validate the performance of those algorithms. In this dissertation, two clustering validity indices named uncertain Silhouette and Order Statistic, are developed for uncertain data. To the best of our knowledge, there is not any clustering validity index in the literature that is designed for uncertain objects and can be used for validating the performance of uncertain clustering algorithms. Our proposed validity indices use probabilistic distance measures to capture the distance between uncertain objects. They outperform existing validity indices for certain data in validating clusters of uncertain data objects and are robust to outliers. The Order Statistic index, in particular, a general form of uncertain Dunn validity index (also developed here), is well capable of handling instances where there is a single cluster that is relatively scattered (not compact) compared to other clusters, or there are two clusters that are close (not well-separated) compared to other clusters. The aforementioned instances can potentially result in the failure of existing clustering validity indices in detecting the correct number of clusters.

Acknowledgements

I would like to express my sincere appreciation to my two co-advisors, Professor Susan Albin and Professor Myong K. Jeong for their guidance and support throughout the course of my Ph.D. studies. They were both patient and supportive mentors who taught me how to think critically and smartly and how to make my research look appealing and relevant. I must also thank my dissertation committee members, Professor Mohsen Jafari, Professor Weihong Guo, and Professor Javier Cabrera for their expertise, support, intellectual insights and time.

I would like to especially thank my wife, Fatemeh, who always believed in me and encouraged me with unwavering love and support. Without her, I could not have done this.

Finally, I must thank my family specially my parents, Sima and Hamid, and my siblings, Ronak and Roozbeh. This past eight years, being far from you, was quite a journey for me. I could not be where I am today, without the love and support of every one of you.

Table of Contents

List of Tables

List of Figures

CHAPTER 1 Introduction

1.1 Overview

In traditional data mining problems with numerical features, each object is associated with several features and each feature is a single point value. These problems are referred to as certain data mining problems. In many practical cases, though, features of each object are represented by multiple observations thus taking into account the probabilistic or uncertain nature of the features. These problems are referred to as uncertain data mining problems. Some common applications where the features of each object are represented by multiple observations are sensor networks, moving object databases and medical and biological databases (Qin et al., 2009a).

The simplest way of dealing with uncertain objects is to consider only one single statistic such as mean or median for each data object. This is equivalent to applying traditional so-called certain data mining algorithms. However, this results in discarding valuable information about each data object that can be crucial to the final outcome. Uncertain data mining algorithms deal with this type of data objects by taking into account a level of uncertainty for each object. That makes the solutions of these algorithms more accurate than the solutions of certain data mining algorithms.

When features of each object consist of many points, we can capture the uncertainty by fitting a probability density function (PDF) to the multiple points. The PDF approach has the advantage of capturing the main characteristics of each object through a few parameters rather than taking into account every single observation within the object.

There has been a lot of interest in developing uncertain data mining techniques in recent years. A good review of published works in four main categories of uncertain data mining problems: classification, clustering, outlier detection and frequent pattern mining, is provided in (Aggarwal and Philip, 2009) and (Liu, 2012). Some examples of the numerous publications in uncertain data mining are: a K-means algorithm and a density-based algorithm for clustering developed in (Chau et al., 2006),(Kriegel and Pfeifle, 2005), a support vector data description (SVDD) algorithm for outlier detection proposed in (Liu et al., 2013), and a frequent pattern mining algorithm developed in (Aggarwal et al., 2009).

For classification; see (Lee et al., 2014), a support vector classifier considering uncertainty for the features as a certain level of noise is developed in (Bi and Zhang, 2005). Decision tree algorithms handling uncertainty in form of PDF are developed in (Qin et al., 2009a) and (Tsang et al., 2011). Naïve Bayesian classifiers for uncertain data are proposed in (Ren et al., 2009) and (B. Qin et al., 2010), while a rule-based classifier is proposed in (Qin et al., 2009b). An associative classifier for uncertain data and a neural network for uncertain data classification are developed in (X. Qin et al., 2010) and (Ge et al., 2010) respectively. A credal classification rule based on the belief functions has been developed in (Liu et al., 2014).

Despite the importance of classification, not enough classification algorithms have been developed. In this dissertation, we present new methodologies for uncertain data classification. At first we develop an uncertain K-nearest neighbor (UKNN) classifier that uses existing Probabilistic Distance Measures (PDM) to measure the distance between objects. Secondly, we propose a new probabilistic distance measure for measuring the distance between an object and a group of uncertain objects. We call this new distance

measure OGPDM as it stands for object-to-group probabilistic distance measure. A weight optimization framework to utilize OGPDM for uncertain data classification is also proposed. Moreover, we introduce definitions for covariance matrix, within, and between scatter matrices for uncertain data objects and use them to extend the Fisher Discriminant Analysis for uncertain data objects.

Clustering validity indices are the main tools for evaluating the quality of formed clusters and determining the correct number of clusters. They can be applied on the results of clustering algorithms to validate the performance of those algorithms. In this dissertation, two clustering validity indices named uncertain Silhouette and Order Statistic, are developed for uncertain data. To the best of our knowledge, there is not any clustering validity index in the literature that is designed for uncertain objects and can be used for validating the performance of uncertain clustering algorithms. Our proposed validity indices use probabilistic distance measures to capture the distance between uncertain objects. They outperform existing validity indices for certain data in validating clusters of uncertain data objects and are robust to outliers. The Order Statistic index in particular, a general form of uncertain Dunn validity index (also developed here), is well capable of handling instances where there is a single cluster that is relatively scattered (not compact) compared to other clusters, or there are two clusters that are close (not well-separated) compared to other clusters. The aforementioned instances can potentially result in the failure of existing clustering validity indices in detecting the correct number of clusters.

1.2 Dissertation outline

This dissertation is organized as follows. Chapter 2 proposes a terminology for modeling uncertain data mining problems. This chapter includes the terminology for

modeling uncertain objects and also uncertain classes, given PDFs. Chapter 3 presents distance-based classifiers for uncertain data objects which includes the proposed uncertain K-nearest neighbor (UKNN) classifier along with the developed object-to-group probabilistic distance measure and also the framework for using the developed distance measure for classification of uncertain data objects. In Chapter 4, measures of scatter and Fisher Discriminant Analysis for uncertain data objects are presented. Chapter 5 presents two developed clustering validity indices for uncertain data objects. Chapter 6 includes the concluding remarks.

CHAPTER 2 Modeling uncertain data objects

This chapter presents notation to describe uncertain objects as well as uncertain classes (groups), i.e., classes (groups) that are composed of uncertain objects. Concepts of variance, covariance, and correlation, for uncertain data objects are also explained.

2.1. Modeling uncertain objects

Uncertain objects may be given in two forms: 1) with multiple points 2) with a probability density function (PDF). Consider K classes of uncertain objects with n_k objects in class k, $k = 1 \dots, K$. We can denote uncertain object i in class k by $\mathbf{O_i^k}$. If uncertain data objects are given with multiple points, $\mathbf{O_i^k}$ denotes a set of points.

If uncertain data objects are given with PDF, it can be written:

$$\mathbf{O_i^k} \sim g^{i,k}\left(\mathbf{x}\middle|\theta^{i,k}\right) \; i = 1,2, \dots, n_k \;\; \text{and } k = 1 \dots, K, \tag{2.1}$$

where $g^{i,k}$ and $\theta^{i,k}$ denote the PDF, and the set of parameters of the PDF for object i in class k.

If we assume multivariate normal distributions, objects can be represented with:

$$\mathbf{O_i^k} \sim \text{MVN}\left(\mathbf{x}|\boldsymbol{\mu}_\mathbf{i}^\mathbf{k}, \Sigma_i^k\right), \; i = 1,2, \dots, n_k \;\; \text{and } k = 1 \dots, K, \tag{2.2}$$

where $\boldsymbol{\mu}_\mathbf{i}^\mathbf{k}$ is the object-mean vector and Σ_i^k is the object-covariance matrix of object i in group k.

Given a new uncertain object, it can also be represented by a PDF: $g(x|\theta_{new})$ where θ_{new} is the set of parameters of the PDF. If we assume multivariate normal distribution for the new object, we would have:

$$\mathbf{O_{new}} \sim \text{MVN}(\mathbf{x}|\boldsymbol{\mu}_\mathbf{new}, \Sigma_{new}), \tag{2.3}$$

where $\boldsymbol{\mu}_{\mathbf{new}}$ and Σ_{new} are the object-mean vector and object-covariance matrix of the new object.

Figure 2.1 is a two-dimensional representation of data depicting 5 blue solid line objects, all from class 1, and one red dashed line new object. The figure assumes each object has a bivariate normal distribution. The dot in the center of each object shows its mean vector, denoted by $\boldsymbol{\mu}_{\mathbf{i}}^{\mathbf{k}}$, $\mathrm{i} = 1, \ldots, 5$, $\mathrm{k} = 1$ for the class, and $\boldsymbol{\mu}_{\mathbf{new}}$ for the new object. The ellipse around the dot shows the contour of the bivariate PDF that has all equal probability of 0.05. The ellipses for the objects in the class are angled to the right indicating that there is a positive correlation between the two features for each object. Similarly, the ellipse for the new object is also angled to the right. The covariance matrices for the class and new objects are labeled $\Sigma_{\mathrm{i}}^{\mathrm{k}}, \mathrm{i} = 1, \ldots, 5$, $\mathrm{k} = 1$ and Σ_{new} respectively. These covariance matrices capture object-correlation.

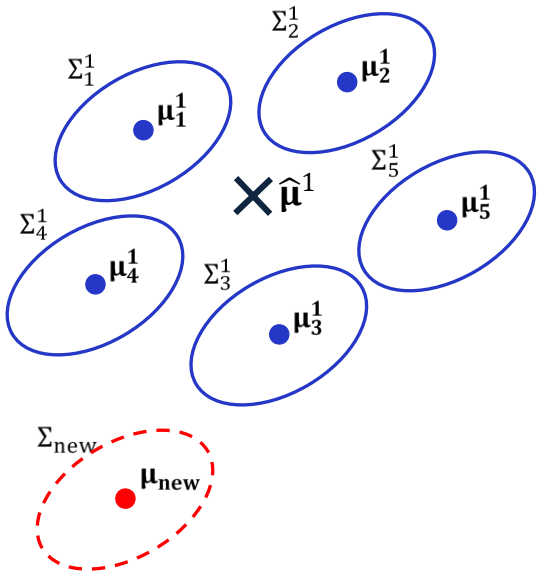


Figure 2.1 A new uncertain object vs class 1 of uncertain objects

2.2. Modeling uncertain classes

We now introduce the concept of characterizing a class in uncertain data mining. Consider the object-mean vectors in the class. These object-mean vectors can be represented by a PDF: $h^k(x|\theta^k)$ where $\theta^{i,k}$ denotes the set of parameters. If we assume multivariate normal distribution for object-mean vectors in class k, we will have:

$$h^k(x|\theta^k) = MVN(\mathbf{x}|\boldsymbol{\mu}^k, \Sigma^k), \tag{2.4}$$

where $\boldsymbol{\mu}^k$ is the class-mean vector for class k and can be estimated by:

$$\hat{\boldsymbol{\mu}}^k = \frac{\sum_{i=1}^{n_k} \boldsymbol{\mu}_i^k}{n_k}, \tag{2.5}$$

and Σ^k is the class-covariance matrix for class k can be estimated by:

$$\hat{\Sigma}^k = \frac{\sum_{i=1}^{n_k} (\boldsymbol{\mu}_i^k - \hat{\boldsymbol{\mu}}^k)(\boldsymbol{\mu}_i^k - \hat{\boldsymbol{\mu}}^k)^t}{n_k - 1}. \tag{2.6}$$

In uncertain data mining problems, there are two types of variance, covariance and correlation. One is the variance for each feature in a class of objects which is called class-variance, covariance among features in a class of objects which is called class-covariance, and correlation among features in a class of objects which is called class-correlation. Class-variance, class-covariance and class-correlation can be estimated from object means. This is the type of variance, covariance and correlation that exists in certain data mining problems as well. There is another type of variance, covariance and correlation too which is unique to the uncertain data problems. That is the variance for each feature within an object, covariance among features within an object and correlation among features within an object. We call this kind of variance, covariance and correlation: object-variance, object-covariance and object-correlation respectively. To our best knowledge, the correlation

among features within objects is not taken into account in most of other uncertain data classification approaches.

Figure 2.2 shows two classes, a red solid line class and a green dashed line class. Each consists of five objects with bivariate normal distributions. The "X" in the center of each class indicates the location of class-mean vector estimate as described in (2.5).

On inspection, it is clear that the red solid line object variances are smaller than the green dashed line object variances; i.e., using the notation in (2.1) the diagonal terms for $\boldsymbol{\Sigma_i^1, i = 1, \ldots, n_1}$ are smaller than the diagonal terms for $\boldsymbol{\Sigma_i^2, i = 1, \ldots, n_2}$. We can also see that the features of objects in the red solid line class are uncorrelated (the ellipses are straight) while the features in the green dashed line class are negatively correlated (ellipses tilt to the left). In other words, there is no object-correlation for the red class while there is negative object-correlation for the green class.

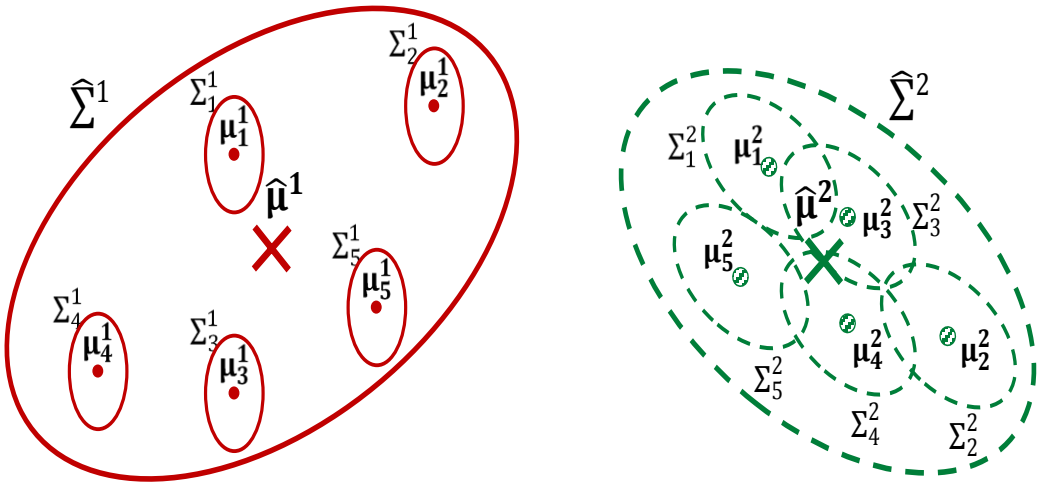


Figure 2.2 Two uncertain data classes with uncertain objects

Further, we can observe another important difference between the red and green classes. Consider now the object mean vectors only. The red object mean vectors are much more spread out than the green. Denoting the covariance matrix of the object mean vectors in class 1 and class 2 by Σ^1 and Σ^2, Figure 2.2 shows that the diagonal terms for Σ^1 are larger in contrast to the smaller diagonal terms for Σ^2. Further, Figure 2.2 shows that the off-diagonal terms for Σ^1 are positive in contrast to the negative off-diagonal terms for Σ^2.

Finally to better understand the definitions of object, class in uncertain data mining problems, consider a generic batch process where the goal is to classify each batch into 2 categories, either conforming or non-conforming to the product specifications (Anzanello et al., 2009, 2012; Xu and Albin, 2006; Zhang et al., 2010; Zhang and Albin, 2007). Suppose it takes say 20 hours for a batch to run and that samples are taken every five minutes by sensors. Thus each feature of the batch is characterized by 240 measurements corresponding to the multiple samples.

One way to deal with this data is to convert it into a certain data mining problem by averaging the samples over each batch (average the 240 measurements) in the training set. Using these averages we can compute the batch-to-batch variances and correlations for each class; i.e., class-variances and class-correlations. However, this approach discards important data. For example, we could not identify a non-conforming batch that is characterized by high variability in the features over its run time or unusual correlations or autocorrelation between features over the batch run.

If instead of using averages we fit the training data for each batch to a joint PDF, say multivariate Normal, we can retrieve the important information such as object-variance and object-correlation that were missing in the averaging approach.

CHAPTER 3 Probabilistic distance-based classifiers for uncertain data objects

In this chapter, we use the concept of probabilistic distance measures (PDM) (Basseville, 1989; Cha, 2007) to develop new uncertain classification algorithms. Probabilistic distance measures are used to measure the distance between two PDFs. They have applications in many fields such as signal processing and communication (Basseville, 1989; Rauber et al., 2008; Zhou and Chellappa, 2004).

This chapter has a few contributions: at first we develop an uncertain K-nearest neighbor (UKNN) classifier that uses existing PDMs to measure the distance between objects. We call this type of distances object-to-object distances as well. UKNN classifies a new object in the same class as the majority of its K closest objects from the training set. This approach results in improvement compared to uncertain naïve Bayesian (UNB) (B. Qin et al., 2010; Ren et al., 2009), as it captures the object-correlation and class-correlation to some extent. In naïve Bayesian classifiers for uncertain data, not only object-correlation is ignored but class-correlation is not taken into account either. Although a possible approach to overcome this issue is to develop Bayesian classifiers (Devijver and Kittler, 1982; Duda et al., 1973); the complexity of estimating the class-conditional joint PDF with multivariate kernel density estimation method makes that approach more complicated. Also, the complexity increases as the dimension gets higher.

The main contribution of the chapter is a new probabilistic distance measure for measuring the distance between an object and a group of uncertain objects. We call our distance measure OGPDM as it stands for object-to-group probabilistic distance measure.

We also propose a weight optimization framework to utilize OGPDM for uncertain data classification.

This proposed approach uses the assumption that the given PDFs describing each object are multivariate Normal. Normal distribution is the favorite in modeling uncertain data with PDF because of its wide range of properties. The concept of the proposed approach is based on classifying the objects to their closest class. One simple possible approach would be to obtain the distance of the object to the center of each class and then classify the object to the class with smaller distance. However, this would not capture any type of variance, class-correlation or object-correlation. Our OGPDM approach is successful in capturing class-correlation and object-correlation even to more extent compared to the proposed uncertain KNN classifier. We will show through various experiments that the OGPDM approach classifies more accurately than the naïve Bayesian classifier, the uncertain KNN classifier, and the certain KNN classifier. The certain classifier here is a KNN classifier that only uses the PDF mean for each object rather than the whole PDF.

Another contribution of our work is to propose a method for simulating uncertain data. Most of the existing work in uncertain data classification is validated through using UCI Machine Learning Repository data sets (Lichman, 2013). UCI repository data sets are real data sets that contain certain data and include various types of data for different data mining applications. The common approach employed by most of the papers is to make UCI repository data uncertain by adding uncertainty to the original data (Ren et al., 2009; Tsang et al., 2011). This might not be the best validation method since the class labels are already set based on certain data and since we should regard uncertainty as a characteristic for each class, adding random uncertainty levels might change the nature of the original data. Our

proposed method does not face this issue and can be used to design various types of experimental scenarios. To have a more complete validation of our proposed approaches, we conduct experiments with the UCI repository data as well.

3.1 Probabilistic distance measures

Probabilistic distances are the distance measures that are defined to capture the distance between two probability density functions. PDMs have many applications in statistics, pattern recognition, communication theory and many other areas (Cover and Thomas, 2012; Csiszár, 1967; Zhou and Chellappa, 2004). In statistics they are mainly used in asymptotic analysis. In pattern recognition probabilistic distance measures such as Chernoff (Chernoff, 1952), Bhattacharyya (Bhattacharyya, 1946), and Lissack-Fu (Lissack and Fu, 1976) are often used to provide a bound on Bayes classification error (Devijver and Kittler, 1982). In communication theory Bhattacharyya distance and KL divergence are used for signal selection. A very good classification of probabilistic distance measures is provided in (Basseville, 1989). The main classes of probabilistic distance measures are introduced as f-divergence family, $\bar{\rho}$ distance, Jensen difference, contrast type measures and spectral distance measures.

Table 3.1 Definition of various probabilistic distance measures

Probabilistic distance measure	Definition
Variational distance	$\frac{1}{2}\int_x \lvert p_1(x) - p_2(x)\rvert dx$
Chernoff distance	$-\log\{\int_x p_1^s(x)p_2^{1-s}(x)dx\}$
Bhattacharyya distance	$-\log\{\int_x p_1^{\frac{1}{2}}(x)p_2^{\frac{1}{2}}(x)dx\}$
Generalized Matusita distance	$[\int_x \left\lvert p_1(x)^{1/r} - p_2(x)^{1/r}\right\rvert^r dx]^{1/r}$
Hellinger distance (Jeffrey-Matusita)	$[\int_x \left\lvert p_1(x)^{1/2} - p_2(x)^{1/2}\right\rvert^2 dx]^{1/2}$

Symmetric KL distance	$\int_x [p_2(x) - p_1(x)] \log(\frac{p_2(x)}{p_1(x)}) dx$

Each of the introduced classes of probabilistic distance measures has properties that are useful depending on the application but in this paper we only focus on the f-divergence family class. The reason is that many of the f-divergence family PDMs have analytical solutions for certain probability distributions. Table 3.1 provides the definition of a few popular probabilistic distance measures which come from the f-divergence family. For Chernoff, $0 \leq s \leq 1$ and can be chosen arbitrarily or depending on the application through optimization. Bhattacharyya is a special case of Chernoff where $s = 0.5$. For Generalized Matusita (Matusita, 1955), $r \geq 1$. The special case of Generalized Matusita, Helinger, is obtained when $r = 2$.

In (3.1), the analytical solutions for Bhattacharyya distance which is one of the f-divergence family PDMs when the two PDFs are multivariate Normal, is shown.

$$d_B = \frac{1}{4}(\boldsymbol{\mu_2} - \boldsymbol{\mu_1})'(\sum_1 + \sum_2)^{-1}(\boldsymbol{\mu_2} - \boldsymbol{\mu_1}) + \frac{1}{2}\log(\frac{|\sum_1 + \sum_2|}{2(|\sum_1|.|\sum_2|)^{\frac{1}{2}}}) \qquad (3.1)$$

Here $\boldsymbol{\mu_1}, \boldsymbol{\mu_2}, \sum_1, \sum_2$ are the parameters of the two multivariate Normal PDFs.

3.2 Uncertain KNN classifier with object-to-object probabilistic distance

In this section we propose our first approach which is based on obtaining the object-to-object probabilistic distance for classification. The approach consists of applying K-nearest neighbor (KNN) classifier using existing probabilistic distance measures. Given the training and test sets, for any single object from the test set (new object), UKNN finds the K closest objects from the training set and assigns the majority class label to the new object.

Since we deal with uncertain objects, in order to obtain the distance between objects, we propose use of probabilistic distance measures. We use K=1 nearest neighbor.

As we model uncertain objects with multivariate Normal PDF, we can utilize the analytical solution of Bhattacharyya distance measure given in (3.1). As the analytical solution of (3.1) utilizes all elements of the object-covariance matrices, this approach should be able to capture the object-correlation and hence results in improved performance with respect to the naïve Bayesian classifier.

3.3 Object-to-group probabilistic distance

In this section we propose a new probabilistic distance measure for measuring the distance between an object and a group of objects and explain how we can utilize it for classifying uncertain objects. We call our distance measure OGPDM which stands for: Object-to-Group Probabilistic Distance Measure. We use the term "Group" rather than "Class" since it implies applications broader than classification which the measure can be utilized for. Also in this section, we propose a method for determining the weight parameters of OGPDM when the goal is classification.

3.3.1 OGPDM distance measure

Given a new object and a group, say class k, our proposed OGPDM can be seen in (3.2):

$$d_{\mathrm{og}}(new,k) = \mathrm{w}_1\left(\boldsymbol{\mu}_{\mathbf{new}} - \widehat{\boldsymbol{\mu}}^{\mathbf{k}}\right)'(\hat{\Sigma}^{\mathrm{k}})^{-1}\left(\boldsymbol{\mu}_{\mathbf{new}} - \widehat{\boldsymbol{\mu}}^{\mathbf{k}}\right) + \mathrm{w}_2 \log\left(\frac{|\Sigma_{\mathrm{new}} + \bar{\Sigma}^{\mathrm{k}}|}{2\left(|\Sigma_{\mathrm{new}}|.|\bar{\Sigma}^{\mathrm{k}}|\right)^{\frac{1}{2}}}\right). \quad (3.2)$$

As it can be noted the measure consists of two components which are linked together through the weight parameters $\mathbf{w_1}$ and $\mathbf{w_2}$, where $\mathbf{w_1} + \mathbf{w_2} = \mathbf{1}$.

The first component, given in (3.3) is:

$$t_{og}^{1}(new,k) = \left(\boldsymbol{\mu_{new}} - \hat{\boldsymbol{\mu}}^{\mathbf{k}}\right)'(\hat{\Sigma}^{\mathrm{k}})^{-1}(\boldsymbol{\mu_{new}} - \hat{\boldsymbol{\mu}}^{\mathbf{k}}). \quad (3.3)$$

As it can be seen from (3.3) the component takes into account the new object-mean vector, class-mean vector and class-covariance matrix. We can use the estimators introduced in (2.5) and (2.6) to obtain the estimates of $\hat{\boldsymbol{\mu}}^{\mathbf{k}}$ and $\hat{\Sigma}^{\mathrm{k}}$. We can recall from those equations that the considered class-covariance matrix is based on object-mean vectors. The component returns smaller distance value as the distance between the new object-mean vector and the class-mean vector gets smaller. Also, the component returns smaller value as the diagonal elements of the class covariance matrix get bigger. The first component provides the advantage of taking into account the class-correlation explicitly which is missing in uncertain Naïve Bayesian and UKNN approach. The second component of the OGPDM is given in (3.4)

$$t_{og}^{2}(new,k) = \log\left(\frac{|\Sigma_{\mathrm{new}} + \bar{\Sigma}^{\mathrm{k}}|}{2(|\Sigma_{\mathrm{new}}|.|\bar{\Sigma}^{\mathrm{k}}|)^{\frac{1}{2}}}\right) \quad (3.4)$$

$\bar{\Sigma}^{k}$ is the average of covariance matrix of objects in class k which is obtained as follows:

$$\bar{\Sigma}^{\mathrm{k}} = \frac{\sum_{\mathrm{i}=1}^{\mathrm{n_k}} \Sigma_{\mathrm{i}}^{\mathrm{k}}}{\mathrm{n_k}} \quad (3.5)$$

This component is inspired from the second component of the analytical solution of Bhattacharyya measure for multivariate Normal PDF as show in (3.1). The component returns smaller distance value, as the covariance matrix of the new object gets more similar to the average covariance matrix of the class. Higher difference in the covariance matrices results in returning higher distance value by the component. The component main advantage compared to the uncertain naïve Bayesian approach is to take into account the object-correlation.

Depending on the application, different ways of obtaining weights should be utilized. For classification, although choosing the weights arbitrarily is the simplest way; it may not result in the best possible classification. The optimal weights should be data-dependent. It means depending on the given data, if there is more difference between class-covariance matrices more weight should be assigned to the first component and if there is more difference between object-covariance matrices more weight should be assigned to the second component. We propose a method to find the weights in the next section.

3.3.2 Determining OGPDM weights for classification

In this section we demonstrate our proposed framework for determining the weights of OGPDM for uncertain data classification. We explain the framework for the two-class ($\mathrm{K} = 2$) classification problem. The way it can be extended for higher values of K will be explained later in this section.

The basic steps are 1) for each object in the training set, obtain the difference between the first terms of its object-to-group PDMs to class 1 and to class 2 using equations (3.3), and also obtain the difference between the second terms of its object-to-group PDMs to class 1 and to class 2 using equation (3.4). 2) standardize the obtained differences for simplicity and avoiding scale issues 3) use the obtained standardized differences for all of the objects in the training set and obtain the optimal weights which form the best hyper-plane that separates the two classes.4) classify new objects or objects in the test set using the hyper-plane.

Now, we explain the above steps in more details. Consider a training data set with two classes, k=1,2, and N uncertain objects. As step 1, for each object i, compute the object-

to-group PDM terms, $t_{og}^{1}(i,k)$ and $t_{og}^{2}(i,k)$, in (3.3) and (3.4). Then, for each object *i*, compute the difference between the first terms $d_{og}^{1}(i)$ as shown in (3.6)

$$d_{og}^{1}(i) = t_{og}^{1}(i,2)\text{-}t_{og}^{1}(i,1), \tag{3.6}$$

and the difference between the second terms $d_{og}^{2}(i)$ as shown in (3.7).

$$d_{og}^{2}(i) = t_{og}^{2}(i,2)\text{-}t_{og}^{2}(i,1). \tag{3.7}$$

In step 2, for simplification and avoiding scale issues, we can convert the difference terms $d_{og}^{1}(i)$ and $d_{og}^{2}(i)$ to the standardized difference terms z_{1i} and z_{2i} in (3.8).

$$\begin{aligned} z_{1i} &= \left(d_{og}^{1}(i) - \frac{\sum_{i=1}^{N} d_{og}^{1}(i)}{N}\right) \Bigg/ \sqrt{\frac{\sum_{i=1}^{N}\left(d_{og}^{1}(i) - \frac{\sum_{i=1}^{N} d_{og}^{1}(i)}{N}\right)^{2}}{N-1}} \\ z_{2i} &= \left(d_{og}^{2}(i) - \frac{\sum_{i=1}^{N} d_{og}^{2}(i)}{N}\right) \Bigg/ \sqrt{\frac{\sum_{i=1}^{N}\left(d_{og}^{2}(i) - \frac{\sum_{i=1}^{N} d_{og}^{2}(i)}{N}\right)^{2}}{N-1}} \end{aligned} \tag{3.8}$$

In step 3, as mentioned before, the goal is to obtain the optimal weights that form the best hyper-plane that can separate the two classes based on the standardized difference terms. Denote the hyper-plane by $w_1 z_1 + w_2 z_2 = 0$. Note that the constant parameter of the hyper-plane is zero because we are using the standardized difference terms.

We propose the use of the Kullback-Leilbler distance (Cha, 2007) to obtain the optimal weights w_1 and w_2. The Kullback-Leilbler distance can be used to determine the separability of the classes in terms of the standardized difference terms. The Kullback-

Leibler distance of the two classes in terms of the standardized difference term $z_j, j = 1,2$ can be denoted by:

$$KL(z_j) = \int_{z_j} p_2(z_j) log(\frac{p_2(z_j)}{p_1(z_j)}) dz_j \quad (3.9)$$
$$j = 1,2$$

where $p_1(z_j), j = 1,2$ is the pdf of the *j-th* standardized difference term for objects in class 1, and $p_2(z_j), j = 1,2$ is the pdf of the *j-th* standardized difference term for objects in class 2.

The weight for the standardized difference term $z_j$can be obtained after normalizing the Kullback-leibler distance as follows:

$$w_j = \frac{KL(z_j)}{\sum_{i=1}^{2} KL(z_i)} \quad (3.10)$$
$$j = 1,2$$

It can be noticed from (3.10) that $w_1 + w_2 = 1$. After obtaining the optimal weights w_1 and w_2,which form the best hyper-plane a new object can be classified in step 4. In this step, the classification rule is: if $w_1 z_{1new} + w_2 z_{2new} \geq 0$, the new object is classified to class 1. This is aligned with the fact that in this case the distance of the new object to class 1 is smaller than its distance to class 2. Conversely, if $w_1 z_{1new} + w_2 z_{2new} < 0$, the rule says classify into class 2 which matches with the fact that the new object distance to class 2 is smaller than its distance to class 1. Having two sets of data, training and test, the optimal hyper-plane can be obtained from the training set and then be used for classifying the objects in the test set.

This framework can be extended to the cases where K is greater than 2. A common procedure to do so is "one class against the rest" where *k* hyperplanes are obtained that

each separates one class from the rest. The classification is performed by combining the results of the k classifiers.

3.4 Experiments using simulated uncertain data

We explain our proposed simulation method for creating uncertain objects and classes. In addition, using the proposed method, we give scenarios to compare the performance of our two proposed approaches with certain KNN and uncertain naïve Bayesian methods.

3.4.1 Simulating uncertain data with multivariate Normal PDF

We propose a new method to simulate uncertain data that can used to validate uncertain classification methods. We assume that both objects and classes have the multivariate Normal distribution as described in equations (2.1), (2.2), and (2.3).

As we explained in the introduction section as well the common approach is to make the UCI repository data uncertain by adding levels of uncertainty. As we mentioned before, this might not be the best validation method since the class labels for the UCI data are already set based on certain data and randomly adding uncertainty might change the nature of the data. Our proposed method does not face this issue. It also enables us to design various types of experimental scenarios to mimic different situations of real data. Another advantage of our simulation method is that we can incorporate both object-correlation and class-correlation in creating uncertain data.

We denote the parameters of the multivariate Normal PDF for object i with: $\boldsymbol{\mu_i^k}$, $\Sigma_i^k; i = 1, \dots, n_k$. The random vectors depicting the object means for class k are generated from the multivariate Normal distribution with parameters $\boldsymbol{\mu^k}, \Sigma^k$ as shown in (3.11).

$$\mathbf{m_i^k} \sim \mathrm{MVN}\left(\boldsymbol{\mu^k}, \Sigma^k\right), \tag{3.11}$$

where $\boldsymbol{\mu_i^k}$ could be considered as a realization of the random vector $\mathbf{m_i^k}$.

The class-covariance matrix Σ^k enables us to model classes where object-mean vectors within a class are correlated.

Object-covariance matrices $\Sigma_i^k;\ i = 1, \dots, n_k;\ k = 1, \dots, K$ are generated using the Inverse Wishart distribution (O'Hagan et al., 2004). Inverse Wishart distribution is used extensively in the literature to simulate real-valued positive definite random matrices (Nydick, 2012). We can consider the object-covariance matrices $\Sigma_i^k;\ i = 1, \dots, n_k;\ k = 1, \dots, K$ as a realization of the random matrices $S_i^k;\ i = 1, \dots, n_k;\ k = 1, \dots, K$ that are generated as follows:

$$S_i^k \sim W^{-1}(\Lambda^k, df^k) * (df^k - p - 1), i = 1, \dots, n_k, \quad k = 1, \dots, K \tag{3.12}$$

where $W^{-1}(\Lambda^k, df^k)$ indicates the inverse Wishart distribution with the covariance matrix $\Lambda^k; k = 1, \dots, K$ which is used as a base matrix for generating random covariance matrices in class k, and the degree of freedom parameter (df) which is used to define the level of deviation from the base matrix. Higher levels of df will result in generating less deviated covariance matrices while lower levels result in more variability.

In the following subsection we compare the performance of our proposed approaches with two existing approaches. In the first approach the uncertain objects are modeled as certain objects by only using the mean vectors of multivariate Normal PDFs and KNN is used to classify. We choose $K = 1$ in our experiments but other values of K can also be tried. In the second approach the uncertain naïve Bayesian classifier proposed in (Ren et al., 2009) is used to classify. We will refer to these methods as "Certain KNN" and "Uncertain Naïve Bayesian Classifier," respectively.

3.4.2 Review of the two existing classification methods: certain KNN and uncertain naïve Bayesian

To classify uncertain objects using certain KNN, each uncertain object is represented by only the mean vector of object multivariate Normal PDF and then KNN with Euclidean distances is used to classify.

Uncertain naïve Bayesian classifier, classifies the new object in the class with the highest posterior probability: $\underset{k}{\operatorname{argmax}} P\left(C_k \middle| g(\boldsymbol{x}|\boldsymbol{\mu_{new}}, \Sigma_{\text{new}})\right), k = 1,2, \dots, K$, where

$$P\left(C_k \middle| g(\boldsymbol{x}|\boldsymbol{\mu_{new}}, \Sigma_{\text{new}})\right) = \frac{P(g(\boldsymbol{x}|\boldsymbol{\mu_{new}}, \Sigma_{\text{new}})|C_k)P(C_k)}{\sum_{k'} P(g(\boldsymbol{x}|\boldsymbol{\mu_{new}}, \Sigma_{\text{new}})|C_{k'})P(C_{k'})}, \tag{3.13}$$

and $P(g(\boldsymbol{x}|\boldsymbol{\mu_{new}}, \Sigma_{\text{new}})|C_k)$ is the probability of observing the PDF of the new object $g(\boldsymbol{x}|\boldsymbol{\mu_{new}}, \Sigma_{\text{new}})$ given the event that the class is k. Further, $P(C_k)$ is the prior probability of C_k $(k = 1,2, \dots, K)$. In uncertain naïve Bayesian classifier (Ren et al., 2009) the features are assumed to be independent and the correlations among them are not taken into account. In other words $P(g(\boldsymbol{x}|\boldsymbol{\mu_{new}}, \Sigma_{\text{new}})|C_k) = \prod_{j=1}^{p} P(g_j\left(\boldsymbol{x}|\mu_{new}^{(j)}, \Sigma_{new}^{(j)}\right)|C_k)$ where $g_j\left(\boldsymbol{x}|\mu_{new}^{(j)}, \Sigma_{new}^{(j)}\right)$ denotes the PDF of the new object along the *j-th* dimension. Therefore, neither class-correlation nor object-correlation can be captured, which are obvious drawbacks of the uncertain-naïve Bayes classifiers.

For uncertain objects modeled with the multivariate Normal PDF, a closed-form solution for the class-conditional probabilities of uncertain naïve Bayesian classifier is given in (Ren et al., 2009) as follows:

$$P(g(\boldsymbol{x}|\boldsymbol{\mu_{new}}, \Sigma_{\text{new}})|C_k) = \prod_{j=1}^{p} \left\{ \sum_{i=1}^{n_k} \frac{\exp(-\frac{1}{2}\left(\frac{\mu_{new}^{(j)} - \mu_i^{k(j)}}{v_{k,i}^{j}}\right)^2)}{n_k v_{k,i}^{(j)} \sqrt{2\pi}} \right\}, \tag{3.14}$$

where $v_{k,i}^{(j)} = \sqrt{h_k^{(j)}.h_k^{(j)} + \Sigma_{new}^{(j)} + \Sigma_i^{k(j)}}$ and $\Sigma_{new}^{(j)}$, and $\Sigma_i^{k(j)}$ respectively denote the variance of the new object along the *j-th* dimension, and the variance of the *i-th* object in class k along the *j-th* dimension. $h_k^{(j)} = 1.06\sqrt{\Sigma_k^{(j)}} n_k{}^{-\frac{1}{5}}$, where $\Sigma_k^{(j)}$ is the variance of the mean of objects in class k along the *j-th* dimension. The power of Bayesian classifiers relies on their ability to incorporate prior information as well. In our experiments we use $P(C_k) = \frac{n_k}{N}, k = 1,2, \dots, K$.

3.4.3 Simulation scenarios

In this section we compare the performance of our proposed approaches with the certain KNN approach and uncertain naïve Bayesian classifier under four simulation scenarios. In the first scenario the performances are compared when only object-correlation exists. The second scenario evaluates the performances when only class-correlation exists. The third scenario compares the performances under the existence of both object-correlation and class-correlation. Finally, the fourth scenario investigates the performances for high dimensions when both types of correlation exist. Class-covariance matrices for the experiments can be expressed in the following form:

$$\Sigma^{\mathrm{k}} = \begin{bmatrix} \mathrm{d_k} & \cdots & \mathrm{o_k} \\ \vdots & \ddots & \vdots \\ \mathrm{o_k} & \cdots & \mathrm{d_k} \end{bmatrix} \tag{3.15}$$

where d_k represents diagonal elements of class k class-covariance matrix and o_k represents off-diagonal elements of class k class-covariance matrix. Hence we consider equal diagonal elements as well as equal off-diagonal elements. Object-covariance matrices also can be expressed as follows:

$$\Lambda^k = \begin{bmatrix} \gamma_k & \cdots & \tau_k \\ \vdots & \ddots & \vdots \\ \tau_k & \cdots & \gamma_k \end{bmatrix} \tag{3.16}$$

where γ_k represents diagonal elements of class k object-covariance matrix and τ_k represents off-diagonal elements of class k object-covariance matrix. Again, we consider equal diagonal elements as well as equal off-diagonal elements.

Each scenario includes ten replicates; where for each, there are two sets of data: training and testing. The training set includes two classes with 1000 p-dimensional objects generated for each class. The testing set also includes two classes with 250 objects generated for each class. Each object from the testing set is classified by applying the classifiers that are firstly trained with the training set.

In all scenarios we assume that the class-mean vectors are: $\boldsymbol{\mu^1} = [0, \dots, 0]'$, $\boldsymbol{\mu^2} = [1, \dots, 1]'$, diagonal elements of class-covariance and object-covariance matrices are: $d_1 = d_2 = 5$, $\gamma_1 = \gamma_2 = 1$ and the degree of freedom parameters are $df^1 = df^2 = 500$. The chosen degree of freedom parameters result in relatively average amount of deviation from the base matrices. In all the tables, the first reported values correspond with the mean of the accuracies of ten replicates and the second reported values (the ones in parenthesis) correspond with the standard deviation of the accuracies of ten replicates.

3.4.2.1 Evaluating the performance when only object-correlation exists

To compare the performance of our proposed approaches on data having only object-correlation, we simulate scenarios for two-dimensional data where the levels of object-correlation are extremely small, very small, and small. The considered object-correlation values are: ±0.05, ± 0.075, and ±0.1. The first column of Table 3.2 describes the scenarios. The next two columns give the off-diagonal parameters for the object-covariance matrices for each class.

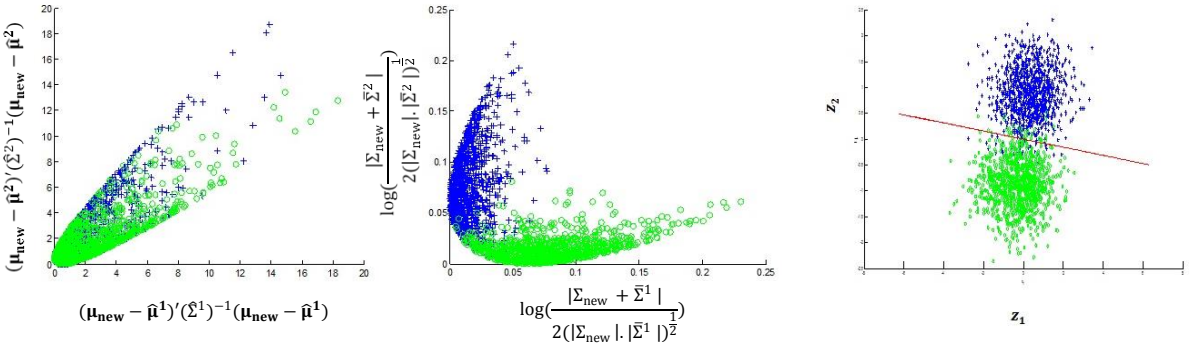
Table 3.2 shows, as we would expect, the certain approach (1NN with Euclidean distance on means) does not perform very well since it does not take into account object-covariance matrix. Uncertain naïve Bayesian classifier takes into account object variances but it also performs poorly since it fails to consider the object-covariance. Uncertain KNN and OGPDM are shown to be able to differentiate the two classes well as the degree of correlation increases.

Table 3.2 Accuracy of the approaches when only object-correlation exists for p=2

Experiment Scenario	Parameter Set $o_1 = o_2 = 0$		Certain KNN	UNB	UKNN	OGPDM
	τ_1	τ_2				
Extremely small correlation	0.05	-0.05	0.555 (0.018)	0.620 (0.021)	0.657 (0.018)	0.867 (0.013)
Very small correlation	0.075	-0.075	0.554 (0.027)	0.637 (0.022)	0.751 (0.021)	0.953 (0.011)
Small correlation	0.1	-0.1	0.536 (0.028)	0.619 (0.025)	0.818 (0.020)	0.987 (0.004)

Although the uncertain KNN approach is also successful in differentiating the two classes; OGPDM gives better results by finding the optimal weights and hence obtaining the best separating hyper-plane. It is noteworthy that although the object-correlations in the investigated scenarios were very small; the proposed approaches were able to achieve high

accuracies in classification. As the object-correlation increases further, the proposed approaches would achieve better classification.



(a) OGPDM component 1 (b) OGPDM component 2 (c) Standardized components

Figure 3.1 Capturing small object-correlation through OGPDM

Figure 3.2 demonstrates the ability of the OGPDM in detecting small object-correlation. The Figure demonstrates the objects in the training set. Objects in class 1 are shown with blue "+" and objects in class 2 with green "o." As it can be seen in Figure 3.1(a), the two classes cannot be differentiated from their class-correlation which is taken into account through OGPDM component 1, however, we can see from Figure 3.1(b) that perfect separation is possible through OGPDM component 2. Figure 3.1(c) demonstrates the separation of the two classes after weight optimization using standardized components. The red line demonstrates the separating hyper-plane.

Table 3.3 OGPDM optimal weights when only object-correlation exists for p=2

Experiment Scenario	w_1	w_2
Extremely small correlation	0.076	0.924
Very small correlation	0.041	0.959
Small correlation	0.018	0.982

The optimal weights for the three scenarios are shown in Table 3.3. As we can see from the table, as the level of object-correlation increases, OGPDM assigns more weight to the second standardized component. This shows the data-dependence nature of our approach.

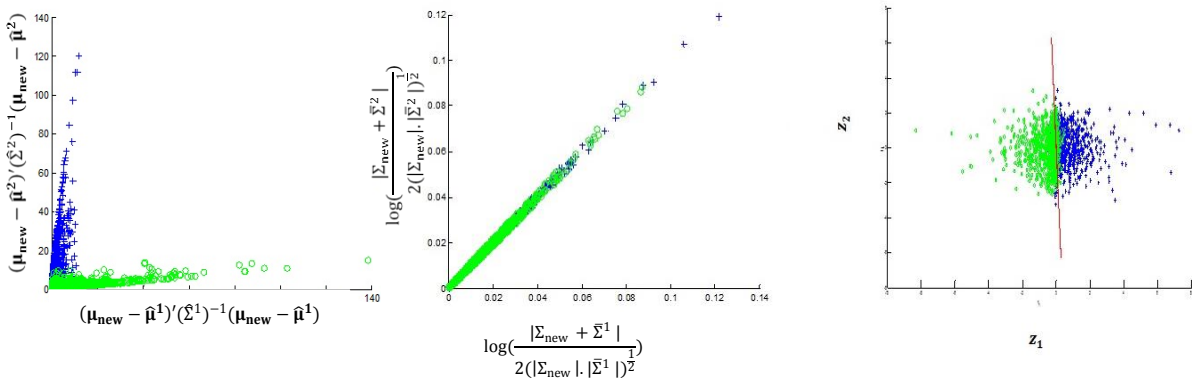
3.4.2.2 Evaluating the performances when only class-correlation exists

In this section, the performance of our proposed approaches having only class-correlation is considered. Again, we consider three experiment scenarios and two classes. The following levels of class-correlation are considered in the scenarios: small correlation, medium, and high. The corresponding correlation values are: 0.1, ± 0.5, and ±0.8. The object-correlation values are set to be equal zero in all scenarios. Class mean vectors and degree of freedom parameters are the same as for the previous section.

Table 3.4 Accuracy of the approaches when only class-correlation exists for p=2

Experiment Scenario	Parameter Set $\tau_1 = \tau_2 = 0$		Certain KNN	UNB	UKNN	OGPDM
	o_1	o_2				
Small correlation	0.5	-0.5	0.542(0.032)	0.613(0.029)	0.553(0.033)	0.615(0.030)
Medium correlation	2.5	-2.5	0.613(0.022)	0.627(0.019)	0.611(0.022)	0.688(0.023)
High correlation	4	-4	0.757(0.019)	0.680(0.028)	0.756(0.014)	0.801(0.021)

As it can be seen from Table 3.4, uncertain naïve Bayesian classifier performs well for smaller class-correlation values but as class-correlation increases, it gets outperformed by other approaches. The certain KNN and the uncertain KNN approaches both achieve higher accuracies as the class-correlation increases. As there is not any object-correlation, the object-to-object correlation information does not provide any advantage over the certain approach. The OGPDM, though, has the advantage over all other approaches since it considers class-covariance matrices and uses the optimal hyper-plane to achieve a higher separation between the two classes.



(a) OGPDM component 1 (b) OGPDM component 2 (c) Standardized components

Figure 3.3 Capturing high class-correlation through the second approach

The ability of the OGPDM approach in detecting high class-correlation can be seen from Figure 3.3. The figure includes the objects in the training set. Again, objects in class 1 are shown with blue "+" and objects in class 2 with green "o". As it can be seen in Figure 3.3(a), the two classes are very well differentiated from their class-correlation which is taken into account through OGPDM component 1, however, we can see from Figure 3.3(b) that separation is not possible through OGPDM component 2 as the object covariance matrices are very similar in the two classes. Figure 3.3(c) demonstrates the separation of the two classes after weight optimization using standardized components.

Table 3.5 OGPDM optimal weights when only class-correlation exists for p=2

Experiment Scenario	w_1	w_2
Small correlation	0.981	0.019
Medium correlation	0.995	0.005
High correlation	0.997	0.003

The optimal weights are shown in Table 3.5. We can see from the table that as the level of class-correlation increases, the OGPDM approach assigns more weight to the first

standardized component. This again shows the data-dependence feature of this proposed approach.

3.4.2.3 Evaluating the performances when a mixture of object-correlation and class-correlation exists

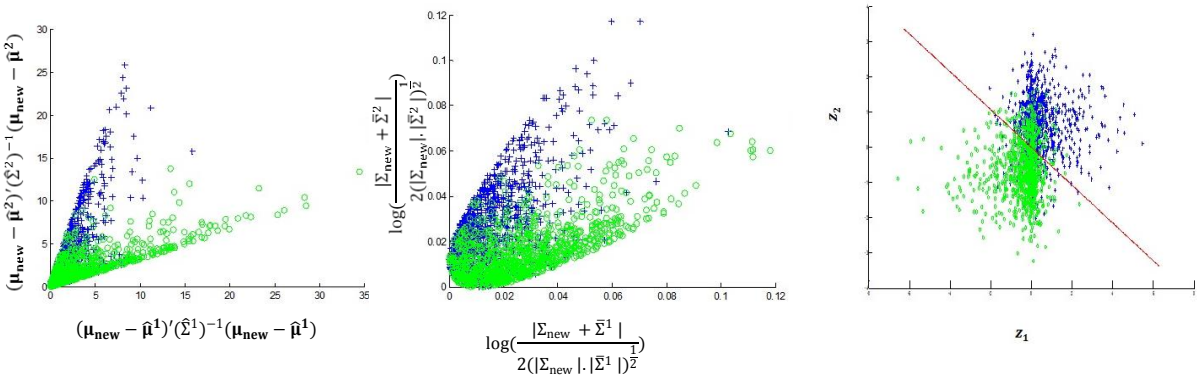
In this section we review a few scenarios in which both type of correlation: object-correlation and class-correlation exist. Extremely small and small levels of object-correlation in companion with medium and high levels of class-correlation are studied. The experiments results in Table 3.6 show the superiority of the OGPDM over all other approaches as it takes into account both types of correlation and uses optimal weights to obtain higher separation. The uncertain KNN is in the second place since it still captures object-correlation and class-correlation. Uncertain naïve Bayesian classifier and certain KNN approaches are next. The advantage of uncertain naïve Bayesian classifier to the certain approach is in considering object-variances. The advantage of the certain KNN approach to the uncertain naïve Bayesian classifier is in its relatively higher sensitivity to the class-correlation.

Table 3.6 Accuracy of the approaches when both object-correlation and class-correlation exist for p=2

Experiment Number	Parameter Set				Certain KNN	UNB	UKNN	OGPDM
	o_1	o_2	τ_1	τ_2				
Medium, Ext small	2.5	-2.5	0.05	-0.05	0.623(0.021)	0.637(0.027)	0.724(0.008)	0.899(0.013)
Medium, small	2.5	-2.5	0.1	-0.1	0.641(0.027)	0.647(0.024)	0.879(0.011)	0.989(0.004)
High, Ext small	4	-4	0.05	-0.05	0.771(0.021)	0.672(0.020)	0.837(0.018)	0.907(0.012)
High, small	4	-4	0.1	-0.1	0.759(0.022)	0.676(0.021)	0.938(0.011)	0.989(0.005)

The ability of the OGPDM in detecting small object-correlation and medium class-correlation can be seen from Figure 3.4. Once again, objects in class 1 are shown with blue "+" and objects in class 2 with green "o." Figure 3.4(a) and Figure 3.4(b) show a relative

differentiation of the two classes based on their object-correlation and class-correlation which are captured through OGPDM components 1 and 2. Figure 3.4(c) demonstrates the separation of the two classes after weight optimization using standardized features.



(a) OGPDM component 1 (b) OGPDM component 2 (c) Standardized components

Figure 3.4 Capturing small object-correlation and medium class-correlation through OGPDM

The optimal weights shown in Table 3.7 give interesting insight about the OGPDM. As both types of correlation exist in the designed scenarios, there exists a balance in the assigned optimal weights to the components.

Table 3.7 OGPDM optimal weights when both object-correlation and class-correlation exist for p=2

Experiment Number	w_1	w_2
Medium, Ext small	0.235	0.765
Medium, small	0.089	0.911
High, Ext small	0.320	0.680
High, small	0.113	0.887

3.4.2.4 Evaluating the performances for higher dimensions

Table 3.8 contains the results of simulation scenarios on two, five and ten dimensional data. The considered simulation parameters are: $o_1 = 0.5$, $o_2 = -0.5$, $\tau_1 = 0.025$, and $\tau_2 = -0.025$. The chosen degree of freedom parameters for these scenarios are $df^1 =$

$df^2 = 300$ which implies more variation in the generated object-covariance matrices. As it can be seen from the table, once again, OGPDM outperforms other approaches in achieving high accuracy. Uncertain KNN outperforms the certain KNN approach because of the existence of object-correlation. For these scenarios, uncertain naïve Bayesian produces good results too as the considered correlation values are relatively small.

Table 3.8 Accuracy of the approaches when both object-correlation and class-correlation exist for p=2, 5, 10

Experiment Number	Certain KNN	UNB	UKNN	OGPDM
p=2	0.551(0.014)	0.625(0.022)	0.573(0.013)	0.711(0.012)
p=5	0.608(0.019)	0.709(0.019)	0.610(0.018)	0.923(0.014)
P=10	0.723(0.011)	0.836(0.014)	0.733(0.017)	0.998(0.017)

3.5 Experiments using real data

In order to have a more complete analysis on the performances of our proposed approaches, we applied them on a few real data sets adopted from the UCI Machine Learning Repository. The chosen data sets are listed in Table 3.9. These data sets are selected from the ones with numerical features. The data sets originally contain only certain objects. That means each object consists of only a single point value.

Table 3.9 Selected data sets from the UCI Machine Learning Repository

Data set	Objects	Features	Classes
Breast Cancer	560	30	2
Ionosphere	350	32	2
Wine	170	13	3
Glass	210	6	6
Blood Transfusion	740	4	2
Heart Statlog	270	13	2
Satellite	4430	36	6
Parkinson’s	190	22	2
Iris	150	4	3
Bank note Authentication	1370	4	2

We can convert each object to an uncertain object in form of multivariate Normal PDF. In this regard, recall (3.1) for PDF of object i in class k, where $\boldsymbol{\mu}_i^k$ was the mean vector and Σ_i^k was the covariance matrix. We consider the original data as $\boldsymbol{\mu}_i^k$ and obtain Σ_i^k as a realization of S_i^k using (3.17):

$$S_i^k = \begin{bmatrix} 0.25 * c_i^k * r_1 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & 0.25 * c_i^k * r_p \end{bmatrix} \qquad (3.17)$$

$$i = 1, \dots, n_k$$

where $r_j, j = 1, .., p$ represents the range of the jth feature of the original certain data and $c_i^k \sim \text{Uniform}(0.95c_k, 1.05c_k)$ where c_k is a controlled parameter defined for class k. As we consider uncertainty as a characteristic of a class, we can consider different c_k values to have different levels of uncertainty for objects of different classes.

We applied 10-fold cross-validation on the selected data sets to obtain the accuracy of the four studied approaches. The accuracy values along with the utilized c_k values for each data set are shown in Table 3.10. The selected c_k values for classes are chosen arbitrarily but we tried very close values for different classes. As it can be seen from the table, for majority of the data sets, OGPDM gives the highest accuracy value. "Bank note authentication" is the only data set where OGPDM stands in the second place with a very small difference compared to the certain KNN and uncertain KNN approaches. The uncertain KNN approach also proves to be a beneficial approach as it outperforms the certain KNN approach in 8 of the 10 selected data sets. It stands in second place above the certain KNN and uncertain naïve Bayesian approaches in seven data sets as well.

Table 3.10 Accuracy of approaches on the selected UCI data sets

Data set	$c_k, k = 1, \ldots, K$	Certain KNN	UNB	UKNN	OGPDM
Breast Cancer	0.05,0.07	0.910(0.043)	0.910(0.043)	0.924(0.048)	0.966(0.019)
Ionosphere	0.05,0.07	0.863(0.062)	0.652(0.070)	0.904(0.045)	0.996(0.008)
Wine	0.05,0.06,0.07	0.763(0.064)	0.951(0.010)	0.959(0.041)	0.989(0.022)
Glass	0.05,0.06,0.07 0.08,0.09,0.1	0.735(0.091)	0.424(0.107)	0.739(0.103)	0.911(0.246)
Blood Transfusion	0.05,0.06	0.702(0.044)	0.756(0.046)	0.757(0.033)	0.983(0.017)
Heart Statlog	0.05,0.07	0.576(0.109)	0.809(0.092)	0.746(0.089)	0.996(0.015)
Satellite	0.05,0.06,0.07 0.08,0.09,0.1	0.905(0.014)	0.722(0.016)	0.851(0.020)	0.954(0.018)
Parkinson's	0.05,0.07	0.853(0.093)	0.717(0.072)	0.924(0.059)	0.975(0.029)
Iris	0.05,0.055,0.06	0.961(0.074)	0.918(0.069)	0.945(0.069)	0.981(0.030)
Bank note Authentication	0.05,0.06	0.999(0.004)	0.834(0.029)	0.999(0.003)	0.997(0.006)

3.6 Conclusion

We proposed two new approaches for classifying uncertain objects modeled with multivariate Normal PDF. Both of the proposed approaches are based on the concept of probabilistic distance measures. The first approach is based on obtaining object-to-object distances. It includes a K-nearest neighbor classifier that can use existing probabilistic distance measures. We choose Bhattacharyya distance measure as the probabilistic distance measure since it has analytical solution for multivariate Normal PDF. This approach was successful in classifying uncertain objects in experiment using both simulated and real data as it proved to be better than the certain KNN approach and uncertain naïve Bayesian classifier in majority of the verified cases.

In order to achieve even better classification performance the second approach was proposed. The second approach is based on the object-to-group distance. In this regard, it uses a proposed probabilistic distance measure called OGPDM. Using OGPDM for classification both object-correlation and class-correlation are captured. The OGPDM

approach provides better classification performance compared to the other approaches as it uses the optimal separating hyper-plane.

CHAPTER 4 Measures of scatter and Fisher Discriminant Analysis for uncertain data

Measures of scatter are extensively used in Statistics particularly in many certain data mining algorithms. They capture the scatter of a group which is either a class or possibly a cluster of data objects. One of the most applied measures of scatter for a group of multivariate data objects is covariance matrix. Covariance matrix conveys valuable information about the level of scatter of data along each dimension and also the scatter with regards to pairs of dimensions. Other types of scatter matrices are within and between scatter matrices. Within scatter matrix captures the scatter within a group of data objects. Between scatter matrix captures the scatter between groups of data objects. Famous applications of within and between scatter matrices are in Analysis of Variance (ANOVA) and also in clustering especially in many clustering validity indices. They are also used in classification algorithms such as Fisher Linear Discriminant Analysis (LDA) (G.McLachlan, 2004).

For uncertain data objects, the existing level of uncertainty can make the scatter of data objects be very different compared to the certain case. Therefore, the scatter measures that are based on certain data objects may not capture the scatter of uncertain data objects properly. Measures of scatter have not been clearly defined for uncertain data objects.

In this chapter, we introduce definitions for covariance matrix, within, and between scatter matrices for uncertain data objects. Our proposed measures of scatter are able to capture the scatter of uncertain objects very better than the measures of scatter for certain data objects.

In addition to introducing measures of scatter for uncertain data objects, in this chapter, we extend Fisher Discriminant Analysis for uncertain data objects. Fisher Discriminant is a well-known classification algorithm. Its popularity rises from the fact that it does not assume any particular probability density function for the distribution of data objects. This is an advantage compared to other Discriminant algorithms such as Linear Discriminant Analysis (LDA) or Quadratic Discriminant Analysis (QDA) that assume normal distributions for data objects.

Fisher Discriminants for certain objects are in two types. One is Fisher Linear Discriminant (LDA) which produces a linear decision boundary and is good when classes of data objects are linearly separable. The other type is Kernel Fisher Discriminant (Mika et al., 1999), which is able to produce both linear and non-linear decision boundaries and therefore works well also for cases where the classes of data objects are not linearly separable.

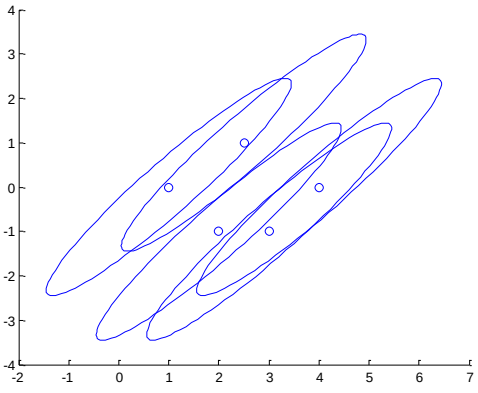
In this chapter, we extend Fisher LDA to the case for uncertain data objects as well and call it Uncertain Fisher LDA. Also we develop Uncertain Kernel Fisher Discriminant for two cases: 1) when the given uncertain objects are in form of multiple points 2) when the uncertain objects are in form of PDF. For each case, we provide an analytical solution to obtain the decision boundary. The developed within and between scatter matrices for uncertain data are used in deriving the solutions for Uncertain Fisher LDA and uncertain Kernel Fisher Discriminants.

4.1 Scatter matrix for uncertain objects

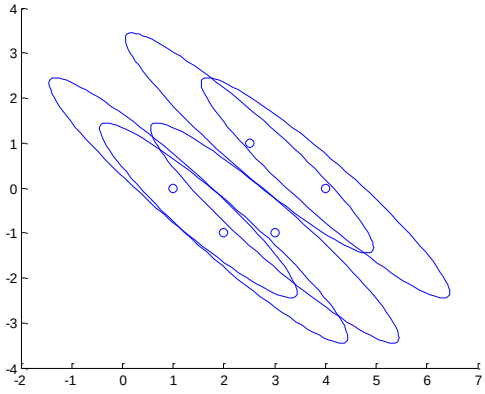
As it was mentioned in the introduction section, the definitions of covariance, within, and between scatter matrices are well-defined for certain data objects. However, for uncertain

data objects, the definitions have not been worked out yet. For uncertain data objects we need new definitions where unlike the certain objects case, here each data object has a level of uncertainty by itself that needs to be taken into account. Our proposed definitions for the three mentioned measures of scatter are provided in this section.

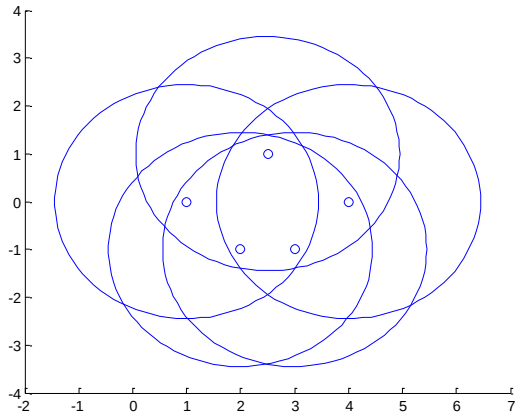
Figure 4.1 shows three groups of uncertain data objects. Uncertain objects in the figures are considered in form of PDF and are shown with dots and ellipses. Dots represent the mean vectors and ellipses represent covariance matrices. In Figure 4.1(a) group of uncertain data objects with positive object-correlation is depicted. Figure 4.1(b) shows a group of uncertain data objects with negative object-correlation. Finally in Figure 4.1(c) a group of data objects with no object-correlation is showed. As we can see, in all three figures there is not much correlation among features considering object-mean vectors only. Considering only object-mean vectors is the procedure that the existing measures of scatter use. Therefore, if the existing measures of scatter are used, the existing correlation among the features in Figure 4.1(a) and Figure 4.1(b) cannot be captured. In addition to capturing correlation structure, the existing scatter matrices fail in capturing the variance.



(a)



(b)



(c)

Figure 4.1 Groups of uncertain data objects with: (a) positive, (b) negative, (c) zero correlation among the features within objects

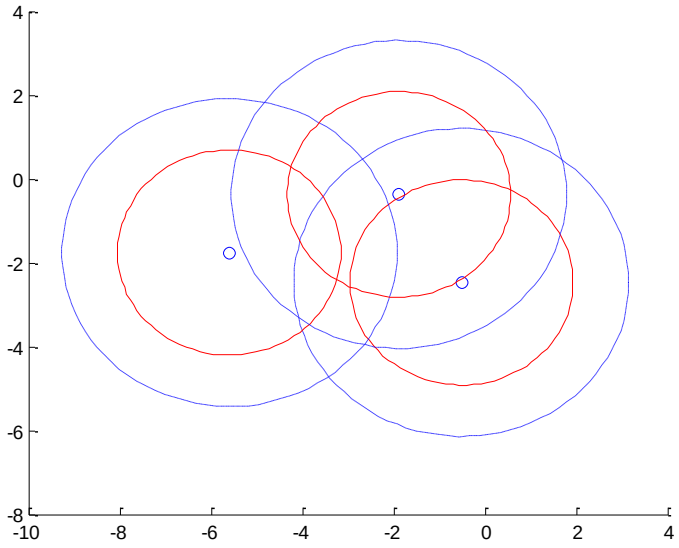


Figure 4.2 Three uncertain data objects with two levels of variance (solid red line and dashed blue line) depicted for each

4.1.1 Covariance matrix for uncertain data objects

As it was already explained, the problem with the existing measures of scatter when used on uncertain data objects is that they don't capture the uncertainty of each data object. In order to overcome this issue, we may consider an uncertain data object as a vector of

random variables that can follow any desired probability density function. For a univariate case, the vector size would be $1 * 1$ but for the multivariate case with p variables, the vector size would be $p * 1$.

Consider a group of n uncertain data objects. Random vector $\mathbf{O_i}$ can be used to represent the $i-th$ object in the group. Although $\mathbf{O_i}$ can follow different type of distributions; we can denote the first moment with $\boldsymbol{\mu_i}$ and the covariance matrix of the object obtained using the first and second moments with $\sum_i$. We call $\boldsymbol{\mu_i}$ as object-mean vector and $\sum_i$ as object-covariance matrix for object i. Denoting the group-mean with $\boldsymbol{\mu}$ we can define the covariance matrix $\sum^U$ for the group of uncertain data objects as follows:

$$\sum^U = E_{\boldsymbol{\mu_i},\sum_i}[E_{\mathbf{O_i}}[(\mathbf{O_i} - \boldsymbol{\mu})(\mathbf{O_i} - \boldsymbol{\mu})^t | \boldsymbol{\mu_i}, \sum_i]], \tag{4.1}$$

where $\widehat{\boldsymbol{\mu}} = \frac{1}{n}\sum_{i=1}^{n} \boldsymbol{\mu_i}$ can be used as the estimate of the group-mean $\boldsymbol{\mu}$. $\sum^U$ can be further expanded as:

$$\sum^U = E_{\boldsymbol{\mu_i},\sum_i}(E_{\mathbf{O_i}}\left[\left((\mathbf{O_i} - \boldsymbol{\mu_i}) + (\boldsymbol{\mu_i} - \boldsymbol{\mu})\right)\left((\mathbf{O_i} - \boldsymbol{\mu_i}) + (\boldsymbol{\mu_i} - \boldsymbol{\mu})\right)^t | \boldsymbol{\mu_i}, \sum_i\right]) =$$

$$E_{\boldsymbol{\mu_i},\sum_i}(E_{\mathbf{O_i}}[(\mathbf{O_i} - \boldsymbol{\mu_i})(\mathbf{O_i} - \boldsymbol{\mu})^t | \boldsymbol{\mu_i}, \sum_i] + E_{\mathbf{O_i}}[(\boldsymbol{\mu_i} - \boldsymbol{\mu})(\boldsymbol{\mu_i} - \boldsymbol{\mu})^t | \boldsymbol{\mu_i}, \sum_i]) = E_{\boldsymbol{\mu_i},\sum_i}([\sum_i +$$

$$(\boldsymbol{\mu_i} - \boldsymbol{\mu})(\boldsymbol{\mu_i} - \boldsymbol{\mu})^t | \boldsymbol{\mu_i}, \sum_i]).$$

Eventually, we can use the following as an unbiased estimate for the covariance matrix of uncertain data objects:

$$\widehat{\sum^U} = \frac{\sum_{i=1}^{n}[\sum_i + (\boldsymbol{\mu_i} - \boldsymbol{\mu})(\boldsymbol{\mu_i} - \boldsymbol{\mu})^t]}{n} = \frac{\sum_{i=1}^{n}(\sum_i)}{n} + \widehat{\sum^C},$$

where $\widehat{\sum^C}$ is the certain covariance matrix estimate which only considers object-mean vectors. Correlation matrix for uncertain objects can be also obtained using the developed covariance matrix:

$$R^U = (\text{diag}(\widehat{\sum^U}))^{-\frac{1}{2}}\widehat{\sum^U}(\text{diag}(\widehat{\sum^U}))^{-\frac{1}{2}} \tag{4.2}$$

Now having a new definition for covariance matrix and correlation matrix of uncertain data objects, we can revisit Figure 4.1. The correlation among the two dimensions in Figure 4.1(a) using the definition of correlation for certain data objects is 0 while the correlation coefficient using the newly defined correlation structure for uncertain data objects is 0.485. Moreover, the variance along the two dimensions using the definition for certain data objects are 1.25 and 0.7, while the corresponding values using the new definition are 2.25 and 1.7. For Figure 4.1(b) the correlation values using the existing and newly defined correlation structures are 0 and -0.485 respectively. The variance values along the two dimensions are also 1.25 and 0.7 for the existing, and 2.25 and 1.7 for the new definition. Finally for Figure 4.1(c) the correlation values would be 0 and 0 for the two definitions. Also variance values are 1.25 and 0.7 for the certain definition versus 2.25 and 1.7 for the uncertain definition. As it is clear from these results, the newly developed definition can better capture the scatter of uncertain data objects.

4.1.2 Total, within, and between scatter matrices for uncertain data objects

Total, within, and between scatter matrices have a lot of applications in clustering, classification and many other fields. Similar to covariance matrix definition that was not well-defined for uncertain data objects, the definition of total within and between scatter matrices also needs to be worked out. Total and within scatter matrices have very similar definitions. For uncertain data we define the total scatter matrix to be: $T^U = \sum_{i=1}^{n} E_{\mathbf{O_i}}[(\mathbf{O_i} - \boldsymbol{\mu})(\mathbf{O_i} - \boldsymbol{\mu})^t]$. Following similar approach to the one used for developing uncertain covariance matrix definition, the formula can be expanded to: $T^U =$

$\sum_{i=1}^{n}[\sum_i + (\boldsymbol{\mu_i} - \boldsymbol{\mu})(\boldsymbol{\mu_i} - \boldsymbol{\mu})^t]$, where again $\boldsymbol{\mu}$ is the group-mean and $\widehat{\boldsymbol{\mu}} = \frac{1}{n}\sum_{i=1}^{n} \boldsymbol{\mu_i}$ can be used as its estimate.

For a one-group problem within scatter matrix and total scatter matrix are the same. For a problem with K groups of uncertain objects where each group consists of n_k objects ($n = \sum_{k=1}^{K} n_k$) within scatter matrix can be defined as:

$$W^U = \sum_{k=1}^{K} W_k^U, \tag{4.3}$$

where W_k^U is the within scatter matrix for group k and can be written as:

$$W_k^U = \sum\nolimits_{i=1}^{n_k} \left[\sum_i^k + \left(\boldsymbol{\mu_i^k} - \boldsymbol{\mu_k}\right)\left(\boldsymbol{\mu_i^k} - \boldsymbol{\mu_k}\right)^t\right], \tag{4.4}$$

where $\boldsymbol{\mu_k}$ is the mean of group k which can be estimated by $\widehat{\boldsymbol{\mu}}_{\boldsymbol{k}} = \frac{1}{n_k}\sum_{i=1}^{n_k} \boldsymbol{\mu_i^k}$. Now that the definitions of total and within scatter matrices are extended for uncertain data objects, the between scatter matrix formulation for uncertain data objects can be obtained as follows:

$$T^U = \sum_{i=1}^{n} E_{\mathbf{O_i}}[(\mathbf{O_i} - \boldsymbol{\mu})(\mathbf{O_i} - \boldsymbol{\mu})^t]$$

$$= \sum_{k=1}^{K}\sum_{i=1}^{n_k} E_{\mathbf{O_i}}\left[\left((\mathbf{O_i} - \boldsymbol{\mu_k}) + (\boldsymbol{\mu_k} - \boldsymbol{\mu})\right)\left((\mathbf{O_i} - \boldsymbol{\mu_k}) + (\boldsymbol{\mu_k} - \boldsymbol{\mu})\right)^t\right]$$

$$= \sum_{k=1}^{K}\sum_{i=1}^{n_k}\left(E_{\mathbf{O_i}}[(\mathbf{O_i} - \boldsymbol{\mu_k})(\mathbf{O_i} - \boldsymbol{\mu_k})^t] + E_{\mathbf{O_i}}[(\boldsymbol{\mu_k} - \boldsymbol{\mu})(\boldsymbol{\mu_k} - \boldsymbol{\mu})^t]\right)$$

$$= \sum_{k=1}^{K}\sum_{i=1}^{n_k}\left[\sum_i^k + \left(\boldsymbol{\mu_i^k} - \boldsymbol{\mu_k}\right)\left(\boldsymbol{\mu_i^k} - \boldsymbol{\mu_k}\right)^t\right] + \sum_{k=1}^{K}\sum_{i=1}^{n_k}(\boldsymbol{\mu_k} - \boldsymbol{\mu})(\boldsymbol{\mu_k} - \boldsymbol{\mu})^t =$$

$$= \sum_{k=1}^{K} W_k^U + \sum_{k=1}^{K}\sum_{i=1}^{n_k}(\boldsymbol{\mu_k} - \boldsymbol{\mu})(\boldsymbol{\mu_k} - \boldsymbol{\mu})^t = W^U + \sum_{k=1}^{K}\sum_{i=1}^{n_k}(\boldsymbol{\mu_k} - \boldsymbol{\mu})(\boldsymbol{\mu_k} - \boldsymbol{\mu})^t = W^U + \sum_{k=1}^{K} n_k(\boldsymbol{\mu_k} - \boldsymbol{\mu})(\boldsymbol{\mu_k} - \boldsymbol{\mu})^t.$$

Knowing that T^U should be composed of within and between scatter matrices, B^U, the between scatter matrix is obtained as:

$$B^U = \sum_{k=1}^{K} n_k(\boldsymbol{\mu_k} - \boldsymbol{\mu})(\boldsymbol{\mu_k} - \boldsymbol{\mu})^t \tag{4.5}$$

The obtained scatter matrix for uncertain data is the same as the between scatter matrix B for certain data.

4.2 Fisher Linear Discriminant Analysis

One of the main approaches for discrimination problem is Fisher Linear Discriminant Analysis (Fisher LDA). Advantage of the Fisher LDA over other discrimination approaches such as Linear Discriminant Analysis (LDA) or Quadratic Discriminant Analysis (QDA) is that it does not assume any particular parametric form for the distribution of classes of data objects. This is unlike the normality assumption which is utilized in LDA and QDA.

The criterion that Fisher LDA uses is based on obtaining the direction for which the projected data objects have the highest ratio of between sum of squares to within sum of squares. In other words, the criterion for Fisher LDA is to find the vector $\mathbf{w}$ that maximizes the function in below:

$$J(\mathbf{w}) = \frac{\mathbf{w}^t B \mathbf{w}}{\mathbf{w}^t W \mathbf{w}}. \tag{4.6}$$

B is the between scatter matrix and can be written as $B = \sum_{k=1}^{K} \sum_{i=1}^{n_k} (\bar{\mathbf{x}}_\mathbf{k} - \bar{\bar{\mathbf{x}}})(\bar{\mathbf{x}}_\mathbf{k} - \bar{\bar{\mathbf{x}}})^t = \sum_{k=1}^{K} n_k (\bar{\mathbf{x}}_\mathbf{k} - \bar{\bar{\mathbf{x}}})(\bar{\mathbf{x}}_\mathbf{k} - \bar{\bar{\mathbf{x}}})^t$ and W is the within scatter matrix and can be written as $W = \sum_{k=1}^{K} W_k = \sum_{k=1}^{K} \sum_{i=1}^{n_k} [(\mathbf{x_i} - \bar{\mathbf{x}}_\mathbf{k})(\mathbf{x_i} - \bar{\mathbf{x}}_\mathbf{k})^t]$ where $\bar{\mathbf{x}}_\mathbf{k}, k = 1,2$ is the mean of objects in class k and $\bar{\bar{\mathbf{x}}} = \frac{1}{n} \sum_{i=1}^{n} \mathbf{x_i}$ is the global mean. It can be proven that vector $\mathbf{w}$ is the

eigenvector of $W^{-1}B$. For the two-class problem $\mathbf{w}$ can be written as $W^{-1}(\bar{\mathbf{x}}_1 - \bar{\mathbf{x}}_2)$. Given the direction vector $\mathbf{w}$, $\mathbf{w}^t\mathbf{x} + b = 0$ would be the decision boundary for classification. There are many ways to obtain the coefficient b but the most conventional one is $b = \frac{\mathbf{w}^t(\bar{\mathbf{x}}_1+\bar{\mathbf{x}}_2)}{2}$. For a new data object $\mathbf{x}_{new}$, if $\mathbf{w}^t\mathbf{x}_{new} + b > 0$, the object is classified to class 1 and conversely, if $\mathbf{w}^t\mathbf{x}_{new} + b < 0$, it would be classified to class 2.

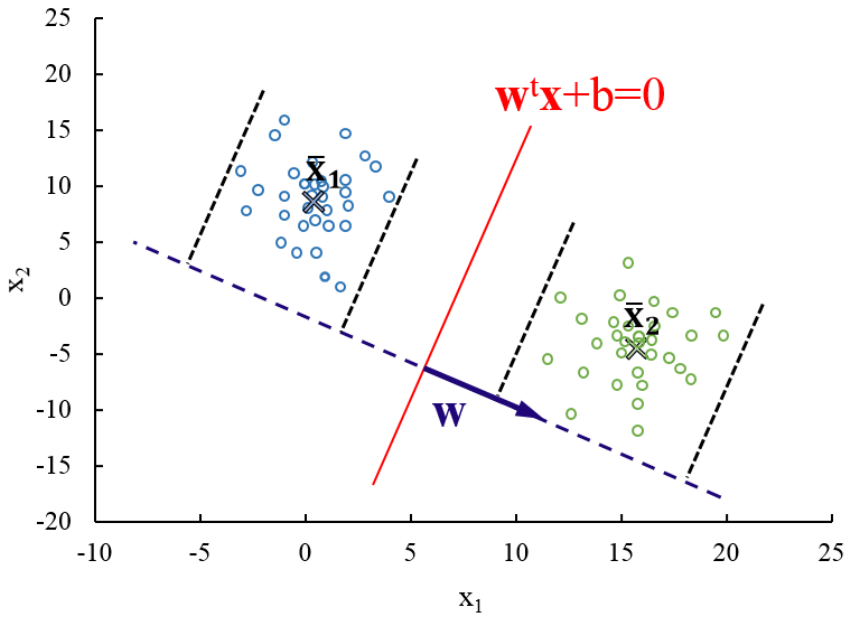


Figure 4.3 Fisher LDA for discriminating two classes of data objects

Figure 4.3 is a demonstration of the direction vector $\mathbf{w}$ along with the linear decision boundary obtained using Fisher LDA for a two-dimensional two-class problem. As we can see the direction vector and the decision boundary are perpendicular.

4.3 Uncertain Fisher Linear Discriminant Analysis

Fisher LDA solution for certain data may not be the best solution for uncertain data as it totally ignores the uncertainty information of objects. In order to overcome this issue, we

propose Uncertain Fisher LDA. Recall the ratio function in (4.6),we rewrite it for uncertain objects as is shown in (4.7).

$$J(\mathbf{w}) = \frac{\mathbf{w}^t B^U \mathbf{w}}{\mathbf{w}^t W^U \mathbf{w}}, \tag{4.7}$$

where $B^U = \sum_{k=1}^{K} n_k(\boldsymbol{\mu_k} - \boldsymbol{\mu})(\boldsymbol{\mu_k} - \boldsymbol{\mu})^t$ and $W^U = \sum_{k=1}^{K} \sum_{i=1}^{n_k} \left[\sum_i^k + (\boldsymbol{\mu_i^k} - \boldsymbol{\mu_k})(\boldsymbol{\mu_i^k} - \boldsymbol{\mu_k})^t\right]$. As we can see although the between scatter matrix is not different than the one in Certain Fisher LDA formulation; the within scatter matrix takes into account uncertain objects covariance matrices and hence the uncertainty information to some extent. Very similar to the case for certain data, vector $\mathbf{w}$ is the eigenvector of $(W^U)^{-1}B^U$, $\mathbf{w}$ can be written as $(W^U)^{-1}(\boldsymbol{\mu_1} - \boldsymbol{\mu_2})$ for a two-class problem, and $\mathbf{w}^t\mathbf{x} + b = 0$ would be the decision boundary for classification. Dealing with uncertain objects, the classification rule needs revision compared to the certain case. If objects are modeled with PDF, given a new object, we can use the following classification rule:

$$\begin{cases} \text{class 1, if } P(\mathbf{w}^t\mathbf{O}_{new} + b > 0) > 0.5 \\ \text{class 2, if } P(\mathbf{w}^t\mathbf{O}_{new} + b < 0) > 0.5 \end{cases}, \tag{4.8}$$

where $\mathbf{O}_{new}$ is a random vector that can follow any type of PDF and is used to model the new object. If objects are modeled with multiple points, given a new object, we can use the following classification rule:

$$\begin{cases} \text{class 1, if } \sum_{t=1}^{l_j^k} I(\mathbf{x_{tj}^k}) > \frac{l_j^k}{2} \\ \text{class 2, if } \sum_{t=1}^{l_j^k} I(\mathbf{x_{tj}^k}) < \frac{l_j^k}{2} \end{cases}, \tag{4.9}$$

where $\mathbf{x}_{\mathbf{tj}}^{\mathbf{k}}$, t=1,…, l_j^k denotes the t-th point within the j-th object in the k-th class. Also, $I(\mathbf{x})$ is an indicator function where $I(\mathbf{x}) = \begin{cases} 1, & \text{if } \mathbf{w}^t\mathbf{x} + b > 0 \\ 0, & \text{if } \mathbf{w}^t\mathbf{x} + b < 0 \end{cases}$. Figure 4.4, shows three situations for two-dimensional two-class problem.

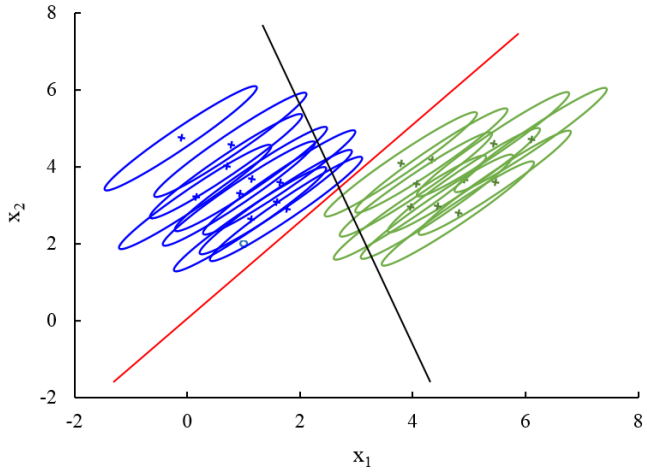


Figure 4.4 Comparing the decision boundaries of Uncertain Fisher LDA with Certain Fisher LDA for two classes of data objects

4.4 Uncertain Kernel Fisher Discriminant Analysis

The developed Uncertain Fisher Discriminant shows improvement compared to the Certain Fisher LDA in regards with the classification accuracy for uncertain objects. However, when classes of uncertain objects are not linearly separable, Fisher LDA would not perform satisfactory. In order to overcome this issue, we use Kernels to obtain non-linear decision boundary. Kernelization consists of transforming data from the original space (linear space) to a higher space (Kernel space). Kernel Fisher Discriminant for certain data objects is developed in (Mika et al., 1999). In this section, we develop Kernel Fisher Discriminant for uncertain data objects and call it “Uncertain Kernel Fisher Discriminant”. We develop

the new algorithm for two forms of uncertain objects: uncertain objects given with multiple points and uncertain objects given with probability density function (PDF).

4.4.1 Uncertain Kernel Fisher Discriminant for objects given with multiple points

Suppose uncertain object are given in form of multiple points where the mean vectors and covariance matrices for objects can be estimated with:

$$\widehat{\boldsymbol{\mu}}_{\mathbf{j}}^{\mathbf{k}} = \frac{\sum_{j=1}^{l_j^k} \mathbf{x}_{\mathbf{tj}}^{\mathbf{k}}}{l_j^k} \text{ and } \widehat{\sum}_j^k = \frac{\sum_t^{l_j^k} (\mathbf{x}_{\mathbf{tj}}^{\mathbf{k}} - \widehat{\boldsymbol{\mu}}_{\mathbf{j}}^{\mathbf{k}})(\mathbf{x}_{\mathbf{tj}}^{\mathbf{k}} - \widehat{\boldsymbol{\mu}}_{\mathbf{j}}^{\mathbf{k}})^t}{l_j^k} \text{ for } j = 1, \dots, n_k,\ k = 1, \dots, K,\ t=1,\dots, l_j^k.$$

Recall the Fisher LDA ratio formula in (4.6). Kernelizing procedure for uncertain data consists of two parts: first part is about kernelizing the numerator of the ratio ($\mathbf{w}^t B^U \mathbf{w}$) and second part is about kernelizing the denominator of the ratio ($\mathbf{w}^t W^U \mathbf{w}$). In the following, we explain the procedure of kernelizing $\mathbf{w}^t B^U \mathbf{w}$ for objects given with multiple points.

Let $\emptyset(\mathbf{x})$ be the map (transformation) of $\mathbf{x}$ into the Kernel space. Recalling the developed formula for the between scatter matrix B^U, for a two-class problem, it can be simplified to $B^U = (\boldsymbol{\mu_1} - \boldsymbol{\mu_2})(\boldsymbol{\mu_1} - \boldsymbol{\mu_2})^t$, we can use $B_\emptyset^U = (\boldsymbol{\mu}_1^\emptyset - \boldsymbol{\mu}_2^\emptyset)(\boldsymbol{\mu}_1^\emptyset - \boldsymbol{\mu}_2^\emptyset)^t$ to denote the between scatter matrix of mapped uncertain data objects, where $\boldsymbol{\mu}_{\mathbf{k}}^\emptyset, k = 1,2$ is the mean of the mapped data objects means in class k**.** As an estimate for $\boldsymbol{\mu}_{\mathbf{k}}^\emptyset$ we can use $\widehat{\boldsymbol{\mu}}_{\mathbf{k}}^\emptyset = \frac{1}{n_k}\sum_{j=1}^{n_k} \emptyset(\widehat{\boldsymbol{\mu}}_{\mathbf{j}}^{\mathbf{k}})$.

Based on the theory of reproducing kernels (Aronszajn, 1950), $\mathbf{w}$ must lie in the span of all samples in the kernel space. This can be written as: $\mathbf{w} = \sum_{i=1}^n \alpha_i \emptyset(\widehat{\boldsymbol{\mu}}_{\mathbf{i}})$ where α_i is a coefficient parameter for object i .

For a two-class problem, by replacing $\mathbf{w}$ and $B_{\emptyset}^{U}$ with $\sum_{i=1}^{n} \alpha_i \emptyset(\hat{\boldsymbol{\mu}}_i)$ and $(\boldsymbol{\mu}_1^{\emptyset} - \boldsymbol{\mu}_2^{\emptyset})(\boldsymbol{\mu}_1^{\emptyset} - \boldsymbol{\mu}_2^{\emptyset})^t$ respectively, we can write:

$$\mathbf{w}^t B_{\emptyset}^{U} \mathbf{w} = \boldsymbol{\alpha}^t M_p \boldsymbol{\alpha}, \tag{4.10}$$

where $\boldsymbol{\alpha}^t = [\alpha_1, \alpha_2, \dots, \alpha_n]$; $M_p = (\mathbf{m_{p1}} - \mathbf{m_{p2}})(\mathbf{m_{p1}} - \mathbf{m_{p2}})^t$; $\mathbf{m_{pk}} = \begin{bmatrix} \frac{1}{n_k} \sum_{j=1}^{n_k} k(\hat{\boldsymbol{\mu}}_1, \hat{\boldsymbol{\mu}}_j^k) \\ \vdots \\ \frac{1}{n_k} \sum_{j=1}^{n_k} k(\hat{\boldsymbol{\mu}}_n, \hat{\boldsymbol{\mu}}_j^k) \end{bmatrix}$.

$k(\mathbf{x}, \mathbf{y})$ is the kernel function for $\mathbf{x}$ and $\mathbf{y}$ and is defined as $k(\mathbf{x}, \mathbf{y}) = \emptyset(\mathbf{x})^t \emptyset(\mathbf{y})$. Polynomial or Gaussian are of the many types that can be used as kernel function.

The detailed procedure of kernelizing $\mathbf{w}^t B^U \mathbf{w}$ for objects given with multiple points, is provided in Appendix I.a. It is noteworthy that, as $B_{\emptyset}^{U}$ is the same as $B_{\emptyset}^{C}$ (for certain data objects), the expanded $\mathbf{w}^t B_{\emptyset}^{U} \mathbf{w}$ is the same as the one developed by (Mika et al., 1999) for Kernel Fisher Discriminant. This formula also holds for the case that data objects are given with PDF except that there is no need to estimate the mean vector and covariance matrix for uncertain objects as they are given.

The second part of the Fisher LDA formula ($\mathbf{w}^t W^U \mathbf{w}$), can be kernelized as follows: Recalling the developed formula for uncertain within scatter matrix W^U, again we can use the estimates of objects mean vector and covariance matrix and rewrite the formula as $W^U = \sum_{k=1}^{K} \sum_{j=1}^{n_k} \left[(\hat{\boldsymbol{\mu}}_j^k - \hat{\boldsymbol{\mu}}_k)(\hat{\boldsymbol{\mu}}_j^k - \hat{\boldsymbol{\mu}}_k)^t + \hat{\sum}_j^k \right]$. $W_{\emptyset}^{U}$ which is the within scatter matrix for the mapped uncertain data objects in the Kernel space can be written as:

$$W_{\emptyset}^{U} = \sum_{k=1}^{K} \sum_{j=1}^{n_k} \left[\left(\emptyset\left(\hat{\boldsymbol{\mu}}_{\mathbf{j}}^{\mathbf{k}}\right) - \hat{\boldsymbol{\mu}}_{\mathbf{k}}^{\emptyset}\right)\left(\emptyset\left(\hat{\boldsymbol{\mu}}_{\mathbf{j}}^{\mathbf{k}}\right) - \hat{\boldsymbol{\mu}}_{\mathbf{k}}^{\emptyset}\right)^{\mathbf{t}} + \frac{\sum_{t=1}^{l_j^k}\left(\emptyset\left(\mathbf{x}_{\mathbf{tj}}^{\mathbf{k}}\right) - \emptyset\left(\hat{\boldsymbol{\mu}}_{\mathbf{j}}^{\mathbf{k}}\right)\right)\left(\emptyset\left(\mathbf{x}_{\mathbf{tj}}^{\mathbf{k}}\right) - \emptyset\left(\hat{\boldsymbol{\mu}}_{\mathbf{j}}^{\mathbf{k}}\right)\right)^{\mathbf{t}}}{l_j^k} \right]. \tag{4.11}$$

Again, using the reproducing Kernel theory where $\mathbf{w} = \sum_{i=1}^{n} \alpha_i \emptyset(\hat{\boldsymbol{\mu}}_{\mathbf{i}})$, we can write:

$$\mathbf{w}^{t} W_{U}^{\emptyset} \mathbf{w} = \boldsymbol{\alpha}^{\mathbf{t}} N_{p1} \boldsymbol{\alpha} + \boldsymbol{\alpha}^{\mathbf{t}} N_{p2} \boldsymbol{\alpha} = \boldsymbol{\alpha}^{\mathbf{t}} N_{p} \boldsymbol{\alpha}, \tag{4.12}$$

where $N_{p1} = \sum_{k=1}^{K} U_{p1k}(I - 1_{n_k}) U_{p1k}{}^{t}$, $N_{p2} = \sum_{k=1}^{K} U_{p2k}(I - 1_{l_j^k}) U_{p2k}^{t}$, and also

$$U_{p1k} = \begin{bmatrix} k\left(\hat{\boldsymbol{\mu}}_{\mathbf{1}}, \hat{\boldsymbol{\mu}}_{\mathbf{1}}^{\mathbf{k}}\right) & \cdots & k\left(\hat{\boldsymbol{\mu}}_{\mathbf{1}}, \hat{\boldsymbol{\mu}}_{\mathbf{n_k}}^{\mathbf{k}}\right) \\ \vdots & \ddots & \vdots \\ k\left(\hat{\boldsymbol{\mu}}_{\mathbf{n}}, \hat{\boldsymbol{\mu}}_{\mathbf{1}}^{\mathbf{k}}\right) & \cdots & k\left(\hat{\boldsymbol{\mu}}_{\mathbf{n}}, \hat{\boldsymbol{\mu}}_{\mathbf{n_k}}^{\mathbf{k}}\right) \end{bmatrix}; \quad U_{p2k} =$$

$$\begin{bmatrix} \sum_{j=1}^{n_k} k\left(\hat{\boldsymbol{\mu}}_{\mathbf{1}}, \mathbf{x}_{\mathbf{1j}}^{\mathbf{k}}\right) & \cdots & \sum_{j=1}^{n_k} k\left(\hat{\boldsymbol{\mu}}_{\mathbf{1}}, \mathbf{x}_{\mathbf{l_j^k j}}^{\mathbf{k}}\right) \\ \vdots & \ddots & \vdots \\ \sum_{j=1}^{n_k} k\left(\hat{\boldsymbol{\mu}}_{\mathbf{n}}, \mathbf{x}_{\mathbf{1j}}^{\mathbf{k}}\right) & \cdots & \sum_{j=1}^{n_k} k\left(\hat{\boldsymbol{\mu}}_{\mathbf{n}}, \mathbf{x}_{\mathbf{l_j^k j}}^{\mathbf{k}}\right) \end{bmatrix};$$

$$1_{n_k} = \begin{bmatrix} 1/n_k & \cdots & 1/n_k \\ \vdots & \ddots & \vdots \\ 1/n_k & \cdots & 1/n_k \end{bmatrix}; 1_{l_j^k} = \begin{bmatrix} 1/l_j^k & \cdots & 1/l_j^k \\ \vdots & \ddots & \vdots \\ 1/l_j^k & \cdots & 1/l_j^k \end{bmatrix}.$$

The detailed procedure of kernelizing $\mathbf{w}^{\mathbf{t}} W^{U} \mathbf{w}$ for objects given with multiple points, is provided in Appendix I.b.

Now, the ratio function in Fisher formula can now be written as:

$$J(\mathbf{w}) = \frac{\mathbf{w}^{\mathbf{t}} B_{U}^{\emptyset} \mathbf{w}}{\mathbf{w}^{\mathbf{t}} W_{U}^{\emptyset} \mathbf{w}} = \frac{\boldsymbol{\alpha}^{\mathbf{t}} M_{p} \boldsymbol{\alpha}}{\boldsymbol{\alpha}^{\mathbf{t}} N_{p} \boldsymbol{\alpha}} = J(\boldsymbol{\alpha}) \tag{4.13}$$

$\boldsymbol{\alpha}$ that maximizes $J(\mathbf{w})$ is obtained as $\boldsymbol{\alpha} = N_p^{-1}(\mathbf{m_1} - \mathbf{m_2})$. Since N_p might be singular, we can use $\boldsymbol{\alpha} = \left(N_p + \lambda I\right)^{-1}\left(\mathbf{m_{p1}} - \mathbf{m_{p2}}\right)$; λ>0 to overcome the possible issues.

The optimal, $\mathbf{w}$ can also be obtained by replacing the optimal $\boldsymbol{\alpha}$ in $\mathbf{w} = \sum_{i=1}^{n} \boldsymbol{\alpha}_i \emptyset(\hat{\boldsymbol{\mu}}_i)$.

4.4.2 Uncertain Kernel Fisher Discriminant for objects given with PDF

Suppose uncertain object are given in form of probability density function (PDF) and the mean vectors and covariance matrices for objects are known: $\boldsymbol{\mu}_{\mathbf{j}}^{\mathbf{k}}$and $\sum_{j}^{k}$ for $j = 1, \dots, n_k$, $k = 1, \dots, K$.

Same as the case for uncertain objects given with multiple points, the kernelization procedure consists of two parts: kernelizing $\mathbf{w}^{t}B^{U}\mathbf{w}$ and kernelizing $\mathbf{w}^{t}W^{U}\mathbf{w}$.

Kernelizing $\mathbf{w}^{\mathbf{t}}B^{U}\mathbf{w}$ for objects given with PDF is the same as the one for Uncertain Kernel Fisher Discriminant for objects given with multiple points. The only difference is that instead of using estimates of mean vectors in the formulation, the actual parameter values are used. Hence, once again:

$$\mathbf{w}^{\mathbf{t}}B_{\emptyset}^{U}\mathbf{w} = \boldsymbol{\alpha}^{\mathbf{t}}M_{d}\boldsymbol{\alpha}, \tag{4.14}$$

where $\boldsymbol{\alpha}^{\mathbf{t}} = [\alpha_1, \alpha_2, \dots, \alpha_n]$; $M_d = (\mathbf{m_{d1}} - \mathbf{m_{d2}})(\mathbf{m_{d1}} - \mathbf{m_{d2}})^{\mathbf{t}}$; $\mathbf{m_{dk}} = \begin{bmatrix} \frac{1}{n_k}\sum_{j=1}^{n_k} k\left(\boldsymbol{\mu}_{\mathbf{1}}, \boldsymbol{\mu}_{\mathbf{j}}^{\mathbf{k}}\right) \\ \vdots \\ \frac{1}{n_k}\sum_{j=1}^{n_k} k\left(\boldsymbol{\mu}_{\mathbf{n}}, \boldsymbol{\mu}_{\mathbf{j}}^{\mathbf{k}}\right) \end{bmatrix}$.

For kernelizing $\mathbf{w}^{\mathbf{t}}W^{U}\mathbf{w}$ for objects given with PDF, once again, recall the developed formula for uncertain within scatter matrix $W^{U} = \sum_{k=1}^{K}\sum_{j=1}^{n_k}\left[\left(\boldsymbol{\mu}_{\mathbf{j}}^{\mathbf{k}} - \boldsymbol{\mu}_{\mathbf{k}}\right)\left(\boldsymbol{\mu}_{\mathbf{j}}^{\mathbf{k}} - \boldsymbol{\mu}_{\mathbf{k}}\right)^{t} + \sum_{j}^{k}\right]$. $W_{\emptyset}^{U}$ which is the within scatter matrix for the mapped uncertain data objects in the Kernel space can be written as:

$$W_{\emptyset}^{U} = \sum_{k=1}^{K}\sum_{j=1}^{n_k}\left[\left(\emptyset\left(\mu_{j}^{k}\right) - \mu_{k}^{\emptyset}\right)\left(\emptyset\left(\mu_{j}^{k}\right) - \mu_{k}^{\emptyset}\right)^{t} + \left(\sum_{j}^{k}\right)^{\emptyset}\right]. \tag{4.15}$$

Using the reproducing kernels theory where $\mathbf{w} = \sum_{i=1}^{n}\alpha_i\emptyset(\boldsymbol{\mu}_{\mathbf{i}})$, we can write:

$$\mathbf{w}^{t}W_{U}^{\emptyset}\mathbf{w} = \boldsymbol{\alpha}^{\mathbf{t}}N_{d1}\boldsymbol{\alpha} + \boldsymbol{\alpha}^{\mathbf{t}}N_{d2}\boldsymbol{\alpha} = \boldsymbol{\alpha}^{\mathbf{t}}N_{d}\boldsymbol{\alpha}, \tag{4.16}$$

where $N_{d1} = \sum_{k=1}^{K} U_{d1k}(I - 1_{n_k}) U_{1k}{}^{t}$, $N_{d2} = \sum_{k=1}^{K} U_{d2k} U_{d2k}^{t}$, and also

$$U_{d1k} = \begin{bmatrix} k(\boldsymbol{\mu_1}, \boldsymbol{\mu_1^k}) & \cdots & k(\boldsymbol{\mu_1}, \boldsymbol{\mu_{n_k}^k}) \\ \vdots & \ddots & \vdots \\ k(\boldsymbol{\mu_n}, \boldsymbol{\mu_1^k}) & \cdots & k(\boldsymbol{\mu_n}, \boldsymbol{\mu_{n_k}^k}) \end{bmatrix}; \; U_{d2k} = \begin{bmatrix} \sum_{j=1}^{n_k} \mathbf{D_{1j}^k}. A_j^k \\ . \\ . \\ . \\ \sum_{j=1}^{n_k} \mathbf{D_{nj}^k}. A_j^k \end{bmatrix}; \text{ and } \mathbf{D_{ij}^k} = \emptyset(\boldsymbol{\mu_i}) J_j^k =$$

$\emptyset(\boldsymbol{\mu_i}).\frac{\partial \emptyset(\mathbf{X})}{\partial \boldsymbol{\mu_j^k}} = \frac{\partial \emptyset(\boldsymbol{\mu_i}).\emptyset(\mathbf{X})}{\partial \boldsymbol{\mu_j^k}} = \frac{\partial \mathbf{K}(\boldsymbol{\mu_i}, \mathbf{X})}{\partial \boldsymbol{\mu_j^k}}$, for $i = 1, \dots, n; j = 1, \dots, n_k$.

The detailed procedure of Kernelizing $\mathbf{w^t} W^U \mathbf{w}$ for objects given with PDF, is provided in Appendix II. Again, similar to the work for the developed formula for multiple points, the ratio function in Fisher formula can now be written as:

$$J(\mathbf{w}) = \frac{\mathbf{w^t} B_U^{\emptyset} \mathbf{w}}{\mathbf{w^t} W_U^{\emptyset} \mathbf{w}} = \frac{\boldsymbol{\alpha^t} M_d \boldsymbol{\alpha}}{\boldsymbol{\alpha^t} N_d \boldsymbol{\alpha}} = J(\boldsymbol{\alpha}). \tag{4.17}$$

Once again, $\boldsymbol{\alpha}$ that maximizes $J(\mathbf{w})$ is $\boldsymbol{\alpha} = N_d^{-1}(\mathbf{m_{d1}} - \mathbf{m_{d2}})$, but for overcoming issues related to singularity of N_d, we can use $\boldsymbol{\alpha} = (N_d + \lambda I)^{-1}(\mathbf{m_{d1}} - \mathbf{m_{d2}})$; $\lambda > 0$. Again, after obtaining the optimal $\boldsymbol{\alpha}$, the optimal $\mathbf{w}$ can also be obtained from $\mathbf{w} = \sum_{i=1}^{n} \boldsymbol{\alpha_i} \emptyset(\boldsymbol{\mu_i})$.

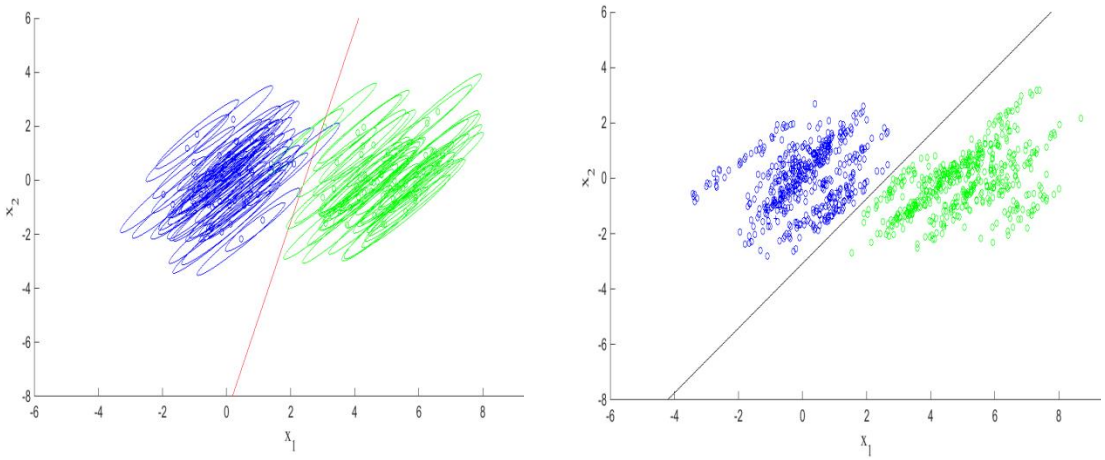
4.5 Simulated examples

In this section first, we provide examples to show the potential of the proposed Uncertain Fisher Discriminants. The examples include two types of uncertain data objects. The ones given with PDF and also the ones given with multiple points. Then, we provide examples to show the potential of the developed Uncertain Fisher Discriminants over the Certain Fisher Discriminants for classification of uncertain data objects.

4.5.1 Uncertain Fisher Discriminants for objects given with PDF and for objects given with multiple points

In this section we present examples to show the potential of the developed Fisher Discriminants for classifying uncertain data objects. For each example, two cases are considered: One where uncertain objects are given with PDF and one where uncertain objects are given with multiple points. The simulation framework to model uncertain objects is the same as the one in (Tavakkol et al., 2017). For each example the decision boundary that is produced by the developed Fisher Discriminants is depicted.

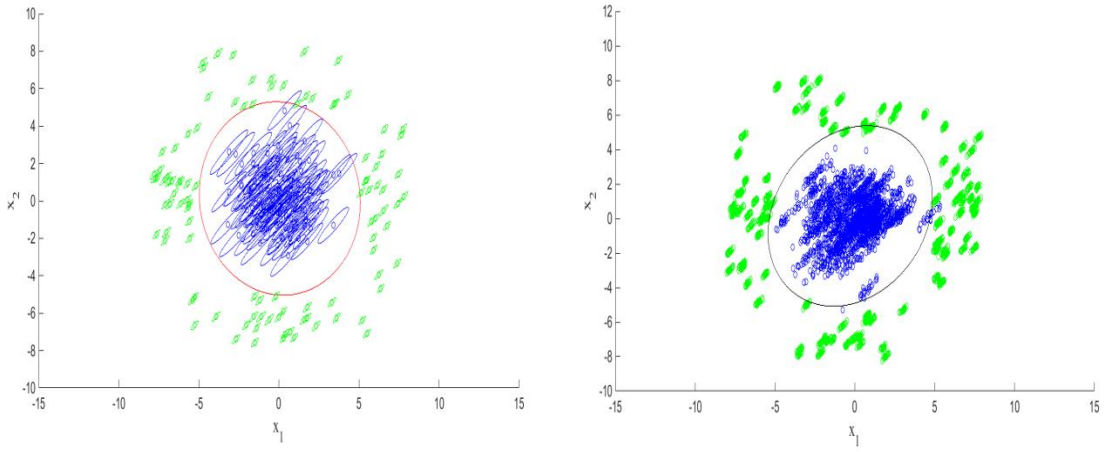
Figure 4.5 shows the linear decision boundaries obtained by the developed Uncertain Fisher LDA. Figure 4.5(a) shows the linear decision boundary for two classes of uncertain objects given with PDF and Figure 4.5(b) shows the linear decision boundary for two classes of uncertain objects given with multiple points. In both Figures the two classes contain objects with positive correlation among their features similar to the case shown in Figure 4.5(a). As we can see the generated decision boundaries for both PDF and multiple points cases are very similar and seem very reasonable for separating the two classes. Comparing this Figure with Figure 4.4 where the generated decision boundary from Certain Fisher LDA is also depicted, we realize the potential of the developed Fisher LDA algorithm for classifying uncertain data objects.



(a) (b)

Figure 4.5 Linear decision boundary obtained by Uncertain Fisher LDA for two classes of positively correlated uncertain objects modeled with a) PDF b) multiple points

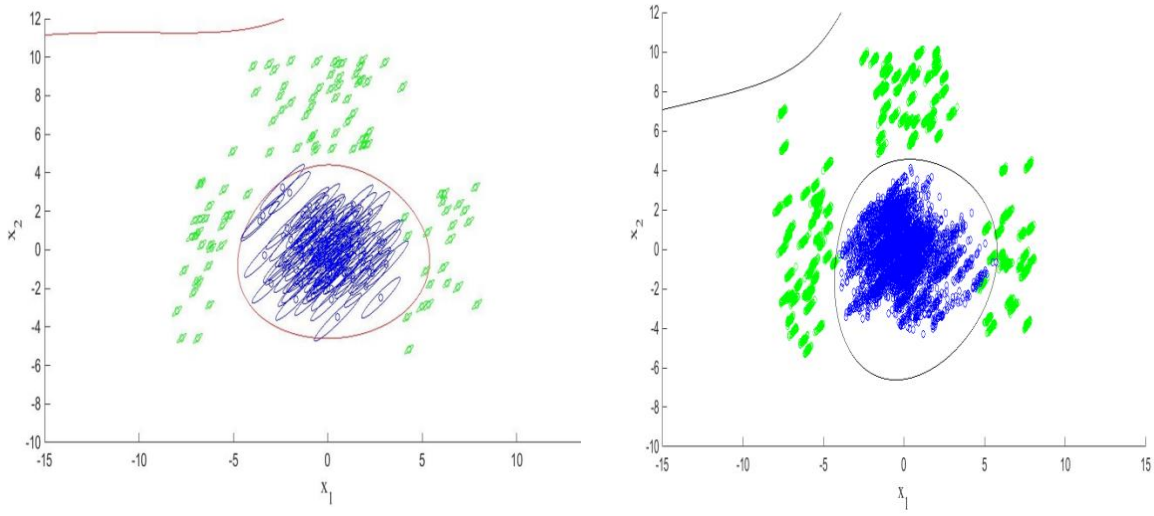
Figure 4.6 shows examples where two classes of uncertain objects are not linearly separable and therefore linear decision boundary does not perform well for separating them. However, second-order polynomials seem appropriate to separate the two classes. Figure 4.6(a) shows the second-order polynomial decision boundary obtained from the developed Uncertain Kernel Fisher Discriminant for objects given with PDF while Figure 4.6(b) shows the second-order polynomial decision boundary obtained from the developed Uncertain Kernel Fisher Discriminant for objects given with multiple points.



(a) (b)

Figure 4.6 Second-order polynomial decision boundary obtained by Uncertain Kernel Fisher Discriminant for two classes of positively correlated uncertain objects modeled with a) PDF b) multiple points

Finally, Figure 4.7 shows examples where third-order polynomial decision boundaries seem reasonable for separating the two classes of data objects. Figure 4.7(a) shows the third-order polynomial decision boundary obtained from the developed Uncertain Kernel Fisher Discriminant for objects given with PDF while Figure 4.7(b) shows the third-order polynomial decision boundary obtained from the developed Uncertain Kernel Fisher Discriminant for objects given with multiple points.



(a) (b)

Figure 4.7 Third-order polynomial decision boundary obtained by Uncertain Kernel Fisher Discriminant for two classes of positively correlated uncertain objects modeled with: a) PDF b) multiple points

As we can see from Figure 4.7, the generated decision boundaries from the developed discriminant for PDF and the one for multiple points are very similar and they all seem very reasonable for separating the generated classes of uncertain data objects.

4.5.2 Performance of uncertain Fisher Discriminants for classification of uncertain data objects

This section gives scenarios that show the potential of the proposed uncertain Fisher discriminants (UFLDA and UKFDA) compared with a few existing classification methods in classifying uncertain objects. The benchmark methods are uncertain K-nearest neighbor (UKNN) (Tavakkol et al., 2017), uncertain naïve Bayesian (UNB) (Ren et al., 2009), uncertain K-means (UK-means) (Xu and Hung, 2011), object-to-group probabilistic distance based classifier (OGPDM) (Tavakkol et al., 2017), Fisher linear discriminant Analysis (FLDA), and kernel Fisher discriminant Analysis (KFDA).

In our simulated scenarios, we consider modeling uncertain objects with the multivariate skew-normal distribution. The skew-normal distribution (Azzalini and Capitanio, 1999; Azzalini and Valle, 1996) is a family of distributions including the normal distribution but it has one more parameter to adjust the skewness. The three parameters of the skew-normal distribution are ξ, Ω, α. ξ is a vector that contains the location parameters. It is very similar to the mean vector parameter in multivariate normal distribution. Ω is a positive-definite matrix which conveys the characteristics of a covariance matrix. α is a vector of parameters that regulates skewness of the distribution. We used the "sn" package in R (Adelchi Azzalini, n.d.) to generate multiple points and model uncertain data objects based on skew-normal distribution.

We considered two sets of simulated scenarios for classifying uncertain data objects. In the first set, we created five scenarios where each contains two classes. The distance between the two classes is different in the scenarios. We labeled the scenarios based on this distance as 1 to 5, where 1 denotes the scenario with smallest distance between location

parameters and 5 is the one with largest. Moreover, for each scenario we considered two sets of data: training set and test set. Training set includes 200 objects (100 class 1 and 100 class 2) and test set includes 100 objects (50 class1 and 50 class 2). Each uncertain object includes 20 generated points. In all five scenarios same Ω and α parameters are used for all objects: $\Omega = \begin{pmatrix} 1 & 0.95 \\ 0.95 & 1 \end{pmatrix}$ and $\alpha = [1000 \quad 0]$. Thus, for object i of class k we can write: $O_i^k \sim SN(\xi^k + \upsilon, \Omega, \alpha)$, $i = 1,...,n_k, k = 1,...,K$, where υ is a random vector where each of its elements follows $uniform(-0.05, 0.05)$. Also parameter ξ^1 in all scenarios is $\xi^1 = [0 \quad 0]$.

Table 4.1 shows the results of classifying uncertain objects in the five scenarios with our proposed UFLDA and also five existing methods: UKNN (with K=3), UNB, UK-means, OGPDM and FLDA. As it can be seen from the table, the performance of all of the classifiers improve as the distance between the location parameters increases. However, in all scenarios, UFLDA performs better classification compared to other methods.

In addition to the first set of scenarios, we considered another set with four more scenarios for classifying uncertain objects but this time the scenarios contain nonlinearly separable uncertain objects. Again, based on the distance between the two classes, we label the four scenarios as: 1, 2, 3, and 4.

Number of data objects in training and test sets and **α** parameter are the same as in the first set of scenarios. Here uncertain data objects in class 1 are generated in two parts: half of them based on ξ_1^1 and the other half based on ξ_2^1. Uncertain data objects in class 2 are generated with $\xi^2 = [0 \quad 0]$. Also for all objects $\Omega = \begin{pmatrix} 1 & -0.95 \\ -0.95 & 1 \end{pmatrix}$.

Table 4.1 Comparing classification accuracies for UKNN, UNB, UK-means, OGPDM, FLDA, and UFLDA classifiers

Scenario	ξ^2	UKNN	UNB	UK-means	OGPDM	FLDA	UFLDA
1	[0.1,0]	0.65	0.68	0.57	0.60	0.63	**0.70**
2	[0.2,0]	0.71	0.91	0.59	0.62	0.89	**0.92**
3	[0.25,0]	0.73	0.91	0.63	0.64	0.91	**0.93**
4	[0.5,0]	0.75	**1.00**	0.69	0.94	**1.00**	**1.00**
5	[0.75,0]	0.77	**1.00**	0.83	**1.00**	**1.00**	**1.00**

Table 4.2 shows the results for the four scenarios of classifying uncertain data objects. For all of the scenarios, classification is performed with UKNN (with K=3), UNB, UK-means, OGPDM, the KFDA with 2nd-order polynomial kernel, as well as our proposed UKFDA with 2nd-order polynomial kernel.

Table 4.2 Comparing classification accuracy UKNN, UNB, UK-means, OGPDM, KFDA, and UKFDA

Scenario	ξ_1^1	ξ_2^1	UKNN	UNB	UK-means	OGPDM	KFDA	UKFDA
1	[-0.25,0]	[0.25,0]	0.73	0.71	0.52	0.80	0.77	**0.84**
2	[-0.5,0]	[0.5,0]	0.75	0.69	0.53	0.95	0.94	**0.97**
3	[-0.75,0]	[0.75,0]	0.75	0.67	0.51	**1.00**	0.93	**1.00**
4	[-1,0]	[1,0]	0.75	0.82	0.56	**1.00**	0.91	**1.00**

The results of Table 4.2 can be interpreted in a similar fashion as the one in Table 4.1. As we can see when the distance between the location parameters of the two classes is small, UKFDA results in better classification accuracy compared to other methods. As the distance between the location parameters increases, performances of all classifiers improve.

We evaluated the performance of our proposed uncertain Fisher discriminants with real data as well. For real data, we collected a weather data set that from the National Center for Atmospheric Research data archive (https://rda.ucar.edu/datasets/ds512.0/) (Jiang et al., 2013). The data set contains average daily temperature and precipitation for 1520 weather stations around the globe in year 2011. We used Köppen-Geiger climate classification (Peel et al., 2007) to assign a climate type to each weather station. The assigned climate types are: polar, cold, temperate, tropical, and dry. Using the data set, we performed five-fold cross-validation to compare the performance of UKNN, UNB, UK-means, OGPDM, FLDA, KFDA, UFLDA, and UKFDA. Table 4.3, reports the average accuracy values for the mentioned classifiers. As it can be seen, our proposed uncertain Fisher discriminants perform reasonably on this dataset. The best result (0.81 accuracy) is produced by UKFDA with 3rd order polynomial kernel. UKFDA with 2nd-order polynomial kernel and OGPDM produce reasonable results as well.

Table 4.3 Comparison of classification accuracies of UKNN, UNB, UK-means, OGPDM, FLDA, UFLDA, KFDA, and UKFDA

UKNN	UNB	UK-means	OGPDM	FLDA	KFDA (2nd order polynomial kernel)	KFDA (3rd order polynomial kernel)	UFLDA	UKFDA (2nd order polynomial kernel)	UKFDA (3rd order polynomial kernel)
0.58	0.53	0.55	0.71	0.52	0.53	0.54	0.55	0.77	**0.81**

4.6 Conclusion

In this chapter, we defined measures of scatter for uncertain data objects. We developed the definition of covariance matrix for uncertain data objects. Within and between scatter matrices were also defined for uncertain objects. Using the developed measures of scatter, we extended the Fisher linear discriminant analysis for uncertain data objects. Also we

developed kernel Fisher discriminants for uncertain data objects. The derivations were developed for two cases: 1) when uncertain objects are given with probability density functions (PDF), 2) when uncertain objects are given with multiple points. We showed through examples that the obtained decision boundaries from our developed uncertain Fisher discriminants seem very reasonable for separating classes of uncertain objects. Also, we evaluated the classification performance on simulated scenarios with uncertain objects modeled with skew-normal distribution and a real data set.

CHAPTER 5 Validity indices for clusters of uncertain data objects

Clustering, one of the main techniques in data mining, falls under the category of unsupervised techniques. Unsupervised techniques work with no class label information provided. Clustering is about organizing the objects in a data set into coherent and contrasted groups or as we call them clusters (Pakhira et al., 2004). The objective is to form clusters so that the objects in the same cluster are close to each other but are far from the objects in other clusters. In other words, the objective of clustering is to form clusters so that they are compact and also well-separated from each other.

For certain data, many popular clustering algorithms exist in the literature (Shin et al. 2012). One of the most well-known is K-means (Chiang et al. 2011; Hartigan and Wong 1979).With the number of clusters known a priori, the K-means algorithm optimizes either by minimizing the within-cluster spread (forming compact clusters), or by maximizing the between-cluster spread (forming separated clusters).

Uncertain data clustering algorithms have been the topic of a few research studies that appear in (Aggarwal and Philip, 2009; Chau et al., 2006; Lee et al., 2007; Gullo et al., 2008b, 2010, 2017, 2013; Kao et al., 2010; Gullo et al., 2008a; Yang and Zhang, 2010; Kriegel and Pfeifle, 2005). A comprehensive survey of uncertain data algorithms which includes clustering algorithms as well is provided in (Aggarwal and Philip, 2009). In (Chau et al., 2006), a K-means clustering algorithm for uncertain data objects is developed which uses the expected distance to capture the dissimilarity between two uncertain objects. It is shown in (Lee et al., 2007) that the uncertain K-means algorithm of (Chau et al., 2006) can be reduced to certain K-means algorithm. A hierarchical clustering algorithm for uncertain

data is proposed in (Gullo et al., 2008b) and (Gullo et al., 2017). Clustering uncertain data using Voronoi diagrams and r-tree index is developed in (Kao et al., 2010). Mixture model clustering of uncertain data objects is investigated in (Gullo et al., 2010, 2013). In (Gullo et al., 2008a; Yang and Zhang, 2010), K-medoids clustering algorithms for uncertain data objects using the expected distance as the distance between the two objects are proposed. In (Jiang et al., 2013; Kriegel and Pfeifle, 2005), density-based clustering algorithms ƑDBSCAN and uncertain DBSCAN with probabilistic distance measures are developed. A K-medoids clustering algorithm that uses probabilistic distance measures for capturing the distance between uncertain objects is also developed in (Jiang et al., 2013). In this paper, we use the uncertain K-medoids clustering algorithm with probabilistic distance measures to evaluate the performance of our proposed clustering validity indices.

There are two important questions that need to be addressed in any clustering problem (Fraley and Raftery, 1998; Halkidi et al., 2001). One is about the actual number of clusters that are present in the data set. And another question is about the validity and goodness of the formed clusters. The answers to these two questions can be obtained by using clustering validity indices. Clustering validity indices are single numerical values that are obtained by incorporating both the compactness and separation of clusters (Pal and Biswas, 1997). When the question is to find the best number of clusters, first, a clustering algorithm such as K-means should be used. The desirable number of generated clusters k, k=1,…,n can be set as an input for the clustering algorithm. Clustering validity indices provide a value for each k, k=1,…,n. Depending on the index, the best number of clusters might be detected as the one that produces the largest or smallest value of the index. Similar to the procedure used to find the best number of clusters, clustering validity indices can be used to evaluate

the goodness of clusters. For any fixed number of clusters, different clustering algorithms might produce different clusters. In these cases also, the best formed clusters can be detected as the ones that produce the largest or smallest index values.

There are many clustering validity indices for certain data objects such as Dunn (Dunn, 1973), Davies-Bouldin (Davies and Bouldin, 1979), Xie-Beni (Xie and Beni, 1991), Silhouette (Rousseeuw, 1987), Calinski-Harbasz (Caliński and Harabasz, 1974), and Pakhira-Bandyopadhyay-Maulik (Pakhira et al., 2005). The first four indices, i.e. Dunn, Davies-Bouldin, Xie-Beni, and Silhouette, are of the most well-known and widely used ones in the literature, and therefore are used for evaluation purposes in this paper. To the best of our knowledge, there is not any clustering validity index in the literature that is designed for uncertain objects modeled with pdf or multiple points and can be used for validating the performance of uncertain clustering algorithms.

In this dissertation, we propose two uncertain clustering validity indices for uncertain data objects: uncertain Silhouette and Order Statistic (OS) index. Our proposed indices not only are superior to existing certain clustering validity indices for validating clusters of uncertain data objects, are also robust to existence of outlier objects. The developed OS index is specifically designed to handle the type of problems where there is either a large dominant compactness value (a very spread cluster), or there is a small dominant separation value (two very close clusters). Those are the two type of problems for which uncertain Dunn index (special case of the OS index which is developed in this paper as well) fails to perform well. Both of the developed indices use probabilistic distance measures to capture the distance between uncertain data objects. Through several experiments, we evaluate the performance of our proposed clustering validity indices over the certain clustering validity

indices. The experiments include three two-dimensional synthetic data sets, a three-dimensional synthetic data set, and a real weather data set. We also show the ability of handling outliers with an experiment with a synthetic data set.

In this chapter, four of the most widely used clustering validity indices for certain data objects are explained in detail: Dunn; Davies-Bouldin; Silhouette; and Xie-Beni. The utilized uncertain K-medoids algorithm is also explained in this chapter. Our proposed uncertain clustering validity indices are explained along with experiments for evaluating the performance of the developed clustering validity indices on synthetic and real data are also presented.

5.1 Clustering validity indices for certain data objects

In this chapter we only consider crisp clusters, i.e., clusters in which objects only belong to one cluster. For this reason, four clustering validity indices that are widely used for crisp certain data are explained in this section. These indices are used for benchmarking. The four indices are Dunn (Dunn, 1973), Davies-Bouldin (Davies and Bouldin, 1979), Xie-Beni (Xie and Beni, 1991), and Silhouette (Rousseeuw, 1987). Dunn, Davies-Bouldin, and Silhouette are indices that are derived based on crisp clusters. Xie-Beni though, is originally derived for fuzzy clusters, i.e., clusters in which objects can belong to more than one cluster. However, its reduced form can be used for crisp clusters. For further discussion on validity indices for crisp and fuzzy clusters, see (Halkidi et al., 2001).

5.1.1 Dunn index

Dunn index is a clustering validity index for clusters of certain data objects. It considers the distance between the two least separated clusters as the separation of the K clusters. It

also considers the compactness of the least compact cluster as the compactness of the *K* clusters. The index is defined in (5.1) for *K* clusters:

$$DU_K = \frac{\min\limits_{\substack{1\le i,j\le K \\ j\neq i}}(dist(C_i, C_j))}{\max\limits_{1\le m\le K}\{diam(C_m)\}} \tag{5.1}$$

where $dist(C_i, C_j)$ denotes the distance between two clusters C_i and C_j and is defined as the distance between the two closest objects of the two clusters:

$$dist(C_i, C_j) = \min_{\boldsymbol{x}\in C_i, \boldsymbol{y}\in C_j}\{d(\boldsymbol{x}, \boldsymbol{y})\} \tag{5.2}$$

and $diam(C_m)$ denotes the diameter of cluster C_m which is used for capturing the compactness of the cluster. The diameter of a cluster, as can be seen in (5.3), is defined as the distance between the two farthest objects in the cluster.

$$diam(C_m) = \max_{\boldsymbol{x},\boldsymbol{y}\in C_m}\{d(\boldsymbol{x}, \boldsymbol{y})\} \tag{5.3}$$

In the above equations, $d(\boldsymbol{x}, \boldsymbol{y})$ denotes the distance between two certain objects $\boldsymbol{x}$ and $\boldsymbol{y}$. $d(\boldsymbol{x}, \boldsymbol{y})$ can be computed using Euclidean distance measure: $d(\boldsymbol{x}, \boldsymbol{y}) = \sqrt{(x_1 - y_1)^2 + \cdots + (x_p - y_p)^2}$ where, $x_j, j = 1, \ldots, p$ denotes the *j*-th dimension of object $\boldsymbol{x}$.

Large values of the Dunn index indicate existence of compact and well-separated clusters.

5.1.2 Davies-Bouldin

Davies and Bouldin (Davies and Bouldin, 1979) propose incorporating separation and compactness of all pairs of certain data clusters C_i and C_j, with $R_{ij}, i, j = 1, \dots, K, i \neq j$, where

$$R_{ij} = \frac{(S_i + S_j)}{d_{ij}} \tag{5.4}$$

and captures both the separation and compactness for the pair of clusters C_i and C_j. S_i and S_j are the components that capture the compactness of certain data clusters C_i and C_j, and d_{ij} captures the distance between the two clusters. The compactness of cluster C_i can be defined as:

$$S_i = \{\frac{1}{n_i} \sum_{\boldsymbol{x_j} \in C_i} \left|\boldsymbol{x_j} - \boldsymbol{z_i}\right|^q\}^{\frac{1}{q}} \tag{5.5}$$

where n_i is the number of objects in cluster C_i and $\boldsymbol{z_i}$ is the centroid of cluster C_i. If q=1, S_i becomes the average Euclidean distance of objects in cluster C_i.to the centroid of the cluster, $\boldsymbol{z_i}$. If q=2, S_i becomes the standard deviation of the distance of objects in the cluster to the cluster center. In general, higher values of S_i indicate less compact and more dispersed clusters.

The distance between clusters C_i and C_j is used to capture the separation of the two clusters. It can be defined as the distance between the centroids of clusters C_i and C_j:

$$d_{ij} = \{\sum_{d=1}^{p} \left|z_{id} - z_{jd}\right|^{w}\}^{\frac{1}{w}} \tag{5.6}$$

where, $z_{id}, d = 1, \dots, p$ denotes the *d*-th dimension of $\boldsymbol{z_i}$. When *w*=1, d_{ij} becomes the "city block" distance and when *w*=2, d_{ij} becomes the Euclidean distance between two centroids. For further discussions on *q* and *w*, see (Davies and Bouldin, 1979). In this chapter, we consider *q*=*w*=2.

Davies-Bouldin uses $\max_{j=1,\dots,K, i\neq j} R_{ij}$ to define R_i for cluster C_i and eventually returns the index value as $DB_K = \frac{1}{K}\sum_{i=1}^{K} R_i$. Small values of the Davies-Bouldin index may indicate more compact and well-separated clusters.

5.1.3 Silhouette

The Silhouette index captures separation and compactness for every single certain object. For *K* clusters the index is defined in (5.7) as follows:

$$SI_K = \frac{1}{n}\sum_{i=1}^{n} \frac{(b_i - a_i)}{\max(a_i, b_i)}. \tag{5.7}$$

In this index, separation and compactness are captured through two components. Compactness for object $\boldsymbol{x_i}$ is captured by component a_i that is defined in (5.8). a_i is defined as the average pairwise distance between object $\boldsymbol{x_i}$ and all objects in the same cluster as object $\boldsymbol{x_i}$ which is denoted by $C_{l_i}, l_i \in \{1, \dots, K\}$.

$$a_i = \frac{1}{\left|C_{l_i}\right|}\sum_{\boldsymbol{y}\in C_{l_i}} d(\boldsymbol{x_i}, \boldsymbol{y}), \tag{5.8}$$

where $|C_{l_i}|$, denotes the number of objects in cluster C_{l_i}.

Separation for object $\boldsymbol{x_i}$ is captured by component b_i that is shown in (5.9). b_i is considered as the separation between object $\boldsymbol{x_i}$ and the closest cluster to it $C_j, C_j \neq C_{l_i}$. The separation between object $\boldsymbol{x_i}$ and cluster C_j is defined as the average pairwise distance between object $\boldsymbol{x_i}$ and all objects in cluster C_j.

$$b_i = \min_j [\frac{1}{|C_j|} \sum_{\substack{\boldsymbol{y} \in C_j \\ C_j \neq C_{l_i}}} d(\boldsymbol{x_i}, \boldsymbol{y})] \tag{5.9}$$

As it can be seen from (5.7), Silhouette, for each object, computes a scaled value of the difference between separation and compactness and eventually, returns the average of the scaled differences over all objects. Higher values of the index imply large separation and also more compactness which are the desirable characteristics of clusters.

5.1.4 Xie-Beni

Xie-Beni index for crisp certain data is defined in (5.10). The index captures compactness by obtaining the mean of squared distances between data objects and their cluster centroids. Separation is captured with the minimum squared distance between cluster centroids.

$$XB_K = \frac{\sum_{i=1}^{K} \sum_{\boldsymbol{x} \in C_{l_i}} d(\boldsymbol{x}, \boldsymbol{z_i})^2}{n \cdot \min\limits_{\substack{i,j=1,\cdots,K \\ i \neq j}} d(\boldsymbol{z_i}, \boldsymbol{z_j})^2} \tag{5.10}$$

For Xie-Beni, smaller values of the index indicate large separation and more compactness.

5.2. Probabilistic distance measures and an uncertain K-medoids clustering algorithm

5.2.1 Measuring the distance between two uncertain objects

In this paper, we utilize probabilistic distance measures (PDM) to capture the distance between two uncertain objects. There are numerous applications for PDMs in many areas such as pattern recognition, communication theory, and statistics (Cover and Thomas, 2012; Csiszar and Körner, 2011; Zhou and Chellappa, 2004). They are also used for estimating the bound on Bayesian classification error, signal selection, and asymptotic analysis (Basseville, 1989; Chernoff, 1952; Devijver and Kittler, 1982). Some of the most well-known probabilistic distance measures are: Variational, Chernoff, Generalized Matusita, Kullback-Leibler, Hellinger, and Bhattacharyya (Basseville, 1989). Hellinger and Bhattacharyya are special cases of Generalized Matusita and Chernoff respectively. Any of these PDMs can be used to capture the distance between two uncertain objects but in this paper, we use Bhattacharyya PDM (Bhattacharyya, 1946), one of the most well-known measures. The definition of Bhattacharyya distance is shown in (5.11):

$$pd_B(\boldsymbol{X}, \boldsymbol{Y}) = -\log(\int_{\boldsymbol{t}} \sqrt{p_{\boldsymbol{X}}(\boldsymbol{t}) p_{\boldsymbol{Y}}(\boldsymbol{t})}\, d\boldsymbol{t}) \tag{5.11}$$

where $p_{\boldsymbol{X}}(\boldsymbol{t})$ and $p_{\boldsymbol{Y}}(\boldsymbol{t})$ denote the pdfs of uncertain objects $\boldsymbol{X}$ and $\boldsymbol{Y}$ and $\boldsymbol{t} \in R^{\boldsymbol{p}}$. If uncertain objects are given in form of multiple points, instead of pdfs, histograms can be built for each object. (5.12) shows the definition of Bhattacharyya PDMwhen objects are given in form of multiple points (Cha, 2007).

$$pd_B(\boldsymbol{X}, \boldsymbol{Y}) = -\ln(\sum_{i=1}^{b} \sqrt{p_{\boldsymbol{X}}{}^{(i)} p_{\boldsymbol{Y}}{}^{(i)}}) \tag{5.12}$$

where $p_{\boldsymbol{X}}{}^{(i)}$ and $p_{\boldsymbol{Y}}{}^{(i)}$ denote the frequency of points in the *i*-th bin for uncertain objects $\boldsymbol{X}$ and $\boldsymbol{Y}$ respectively. In the equation, b denotes the number of bins.

One of the main advantages of using Bhattacharyya PDM is when uncertain objects follow multivariate normal distributions, Bhattacharyya yields an analytical solution for the PDM between the two objects as shown in (5.13):

$$pd_B(\boldsymbol{X}, \boldsymbol{Y}) = \frac{1}{4}(\boldsymbol{\mu_X} - \boldsymbol{\mu_Y})'(\Sigma_X + \Sigma_Y)^{-1}(\boldsymbol{\mu_X} - \boldsymbol{\mu_Y}) + \frac{1}{2}\log(\frac{|\Sigma_X + \Sigma_Y|}{2(|\Sigma_X||\Sigma_Y|)^{\frac{1}{2}}}), \tag{5.13}$$

where $\boldsymbol{X} \sim MVN(\boldsymbol{\mu_X}, \Sigma_X)$ and $\boldsymbol{Y} \sim MVN(\boldsymbol{\mu_Y}, \Sigma_Y)$. Here, $\boldsymbol{\mu_X}$ and $\boldsymbol{\mu_Y}$ are means, and Σ_X and Σ_Y are covariance matrices.

5.2.2 Uncertain K-medoids clustering algorithm

Different uncertain K-medoids clustering algorithms have been proposed in the literature. Uncertain K-medoids algorithms that use the expected distance to capture the dissimilarity between two uncertain objects are developed in (Gullo et al., 2008a; Yang and Zhang, 2010). In (Jiang et al., 2013), an uncertain K-medoids algorithm that uses PDMs to capture the distance between uncertain objects, is proposed. In this chapter, we use the latter algorithm and use Bhattacharyya as the PDM. The steps of the uncertain K-medoids algorithm are as follow:

Step 1: pick *K* initial uncertain objects (medoids) randomly. Form clusters by assigning each object to the cluster for which the probabilistic distance between the object and the cluster medoid is smallest.

Step 2: Obtain the new medoids, $m_k, k = 1, \dots, K$, as follow:

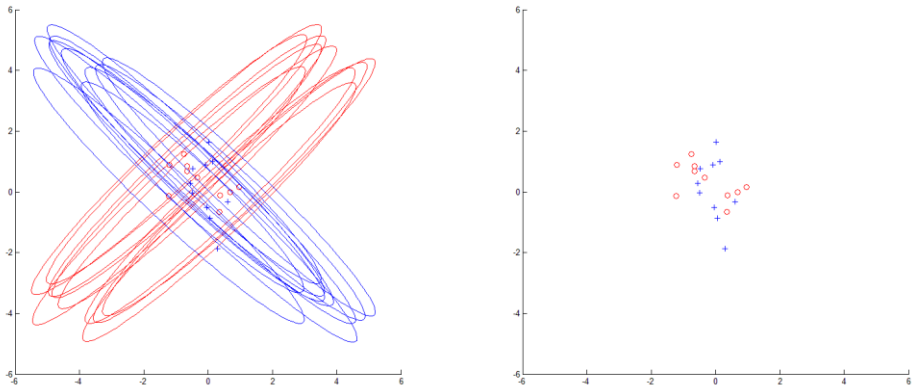
$$m_k = arg \min_{\boldsymbol{X_i} \in C_k} \sum_{\boldsymbol{X_j} \in C_k \setminus \{\boldsymbol{X_i}\}} pd_B(\boldsymbol{X_i}, \boldsymbol{X_j}) \tag{5.14}$$

where, $pd_B(\boldsymbol{X_i}, \boldsymbol{X_j})$ denotes the Bhattacharyya probabilistic distance between $\boldsymbol{X_i}$ and $\boldsymbol{X_j}$.

Step 3: Using the new medoids, re-assign each object to the cluster of its nearest medoid. Repeat Step 2 and Step 3 until there is no change in the clusters.

5.3 The proposed uncertain clustering validity indices

In this section we explain the reason uncertain data objects require their own clustering validity indices through an example. Fig. 5.1(a) shows a two-dimensional example where there are two clusters of uncertain data objects. Objects in both clusters are in form of bivariate normal pdfs and are represented by ellipses. But objects in one cluster (shown in red) have positive correlation among their two dimensions, while objects in the other cluster (shown in blue) have negative correlation. Applying uncertain K-medoids clustering algorithm with *K*=2 on the objects, the two clusters are detectable.



(a) (b)

Figure 5.1 Two clusters of uncertain data a) each uncertain object shown with its whole pdf. b) each uncertain object from (a) shown with its mean only.

In order to find the correct number of clusters of this example (i.e. two), a clustering validity index is needed. If the clustering validity indices for certain data objects that only use the object means, are used, the results would not be desirable and one cluster would be preferred to two clusters. The reason can be seen in Fig. 5.1(b), where only the object means are shown and it is impossible to distinguish between the red and blue clusters. Clustering validity indices designed for uncertain data objects should prefer two clusters over a single cluster in this example. We show in the experiments section that our developed uncertain clustering validity indices are well capable of doing so.

5.3.1 Uncertain Silhouette

Our first proposed cluster validity index for uncertain data objects is called uncertain Silhouette index. The definition of the uncertain Silhouette is shown in (5.15).

$$USI_K = \frac{1}{n}\sum_{i=1}^{n}\frac{(ub_i - ua_i)}{\max(ua_i, ub_i)} \tag{5.15}$$

where ua_i denotes the compactness and ub_i denotes the separation for uncertain object $\boldsymbol{X_i}$. The definitions of ua_i and ub_i are shown in (5.16) and (5.17) respectively. As it can be seen from (5.16), similar to the case for certain data as in (5.8), compactness of an object $\boldsymbol{X_i}$ is defined as the average pairwise distance between the object $\boldsymbol{X_i}$ and all objects in the same cluster as object $\boldsymbol{X_i}$. The main difference between ua_i and a_i (compactness component of Silhouette index for certain data objects) is that in ua_i PDMs are used to better capture the distance between uncertain objects, while in a_i distance measures for certain data objects such as Euclidean are used.

$$ua_i = \frac{1}{|C_{l_i}|} \sum_{\boldsymbol{Y} \in C_{l_i}} pd(\boldsymbol{X_i}, \boldsymbol{Y}) \tag{5.16}$$

As it can be seen from (5.17), similar to the case for the original Silhouette index, ub_i is considered as the separation between object $\boldsymbol{X_i}$ and the closest cluster to it $C_j, C_j \neq C_{l_i}$. The separation between object $\boldsymbol{X_i}$ and cluster C_j is defined as the average pairwise distance between object $\boldsymbol{X_i}$ and all objects in cluster C_j. Again, the main difference between ub_i and b_i is that in ub_i PDMs are used to capture the distance between objects, while in b_i distance measures for certain data objects are used.

$$ub_i = \min_j \left(\frac{1}{|C_j|} \sum_{\substack{\boldsymbol{Y} \in C_j \\ C_j \neq C_{l_i}}} pd(\boldsymbol{X_i}, \boldsymbol{Y}) \right) \tag{5.17}$$

In this, we use Bhattacharyya as the PDM for computing the uncertain Silhouette index. Same as the original Silhouette, the optimal setting is the one that produces the largest index value and possibly the one that has the most compact and well-separated clusters.

5.3.2 OS index

In this section we propose a new clustering validity index for uncertain data objects, named Order Statistic (OS). The OS index can be considered as a general form of uncertain Dunn index, which is also developed in this paper. The OS index is composed of two components for capturing separation and compactness of clusters. It considers the average of r (r >1) smallest inter-cluster distances for separation, and also the average of r (r >1) largest intra-cluster distances for compactness. This enables the index to correctly detect the correct number of clusters in cases where there is either a very scattered cluster, or two very close clusters. The aforementioned cases are the ones for which uncertain Dunn index (r =1) will fail in detecting the correct clusters. We propose r=K-1 and find it a reasonable choice for r. The OS index is shown in (5.18).

$$OS = \frac{\sum_{i=1}^{r} sp_{(i)} / r}{\sum_{j=K-r+1}^{K} cp_{(j)} / r} \tag{5.18}$$

where $sp_{(i)}, i = 1, \dots, \frac{K(K-1)}{2}$ is the i-th smallest order statistic of inter-cluster distances. The first order statistic of inter-cluster distances is $sp_{(1)} = \min_{\substack{1 \le C_i, C_j \le K \\ C_i \neq C_j}} [dist(C_i, C_j)]$. Here, for $dist(C_i, C_j)$, which denotes the distance between clusters C_i and C_j, we propose the average of s smallest pairwise probabilistic distances between objects in cluster C_i and objects in cluster C_j. Capturing the distance between two clusters in this fashion has the advantage of being more robust to the existence of outlier values.

$cp_{(j)}, j = 1, \dots K$, is the j-th smallest order statistic of intra-cluster distances. The K-th order statistic of intra-cluster distances is $cp_{(K)} = \max_{1 \le C_m \le K} [diam(C_m)]$. Here, $diam(C_m)$

denotes the diameter of cluster C_m and basically captures the compactness of the cluster. For $diam(C_m)$, we propose the average of *t* largest pairwise probabilistic distances between objects in cluster C_m. Capturing the diameters of clusters in this fashion has the advantage of being more robust to the existence of outlier values as well.

In the experiment section, we try two settings for the parameters *s* and *t* of the OS: 1) *s*=*t*=3, and 2) *s*=*t*=5. If we choose *r*=*s*=*t*=1, the OS index will reduce to an index that we call uncertain Dunn. Uncertain Dunn index is defined in (5.19).

$$UDU_K = \frac{sp_{(1)}}{cp_{(1)}} = \frac{\min\limits_{\substack{1\le i,j\le K \\ j\neq i}}(dist(C_i, C_j))}{\max\limits_{1\le m\le K}\{diam(C_m)\}} \tag{5.19}$$

In this index, $dist(C_i, C_j)$ and $diam(C_m)$ are defined based on (5.20) and (5.21).

$$dist(C_i, C_j) = \min_{\boldsymbol{X}\in C_i, \boldsymbol{Y}\in C_j}\{pd(\boldsymbol{X}, \boldsymbol{Y})\} \tag{5.20}$$

$$diam(C_m) = \max_{\boldsymbol{X}, \boldsymbol{Y}\in C_m}\{pd(\boldsymbol{X}, \boldsymbol{Y})\} \tag{5.21}$$

(5.19-5.21), can be compared with (5.1-5.3) for certain Dunn index. Large values of the index indicate existence of compact and well-separated clusters.

One of the drawbacks of this uncertain Dunn index is its sensitivity to outlier values. Existence of outliers can highly affect (5.20) and (5.21), and therefore the whole index. Another drawback of the uncertain Dunn index is its poor performance in the presence of either dominant small separation or large compactness values.

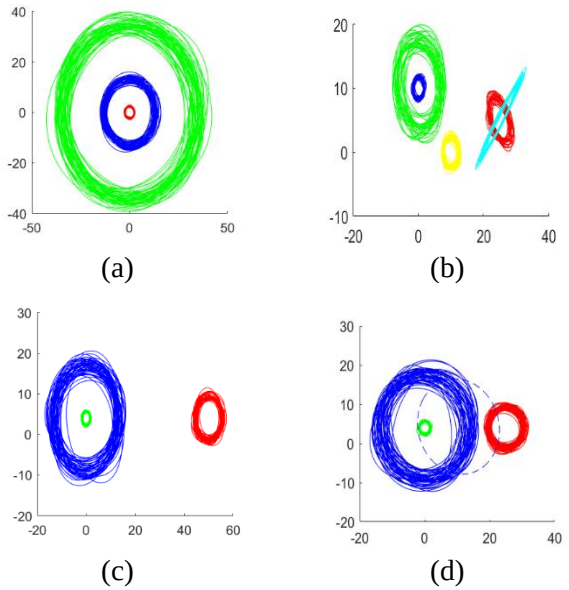
In the experiments section we show the capability of the OS and Silhouette indices over uncertain Dunn in overcoming these drawbacks through several experiments.

5.4 Experiments

The effectiveness of our proposed uncertain clustering validity indices is demonstrated through experiments on the following data sets: four two-dimensional synthetic data sets, a three-dimensional synthetic data set, and the weather data set. Uncertain objects in each synthetic data set are modeled with multivariate normal distribution. Performance of Dunn, Davies-Bouldin, Xie-Beni, Silhouette, uncertain Dunn, uncertain Silhouette, and OS with different parameters are compared.

5.4.1 Two dimensional synthetic data sets

We conducted experiments on four two-dimensional data sets named SD1, SD2, SD3, and SD4. For each data set, different number of clusters was generated and each cluster contained 50 uncertain objects. Fig. 5.2 shows the generated clusters for each data set. For SD1, SD3, SD4, three, and for SD2, five clusters were generated. SD3 and SD4 are very similar, except that for SD4 there exist a major outlier (shown with dashed ellipse). This can be observed by comparing Fig. 5.2(c) and Fig. 5.2(d).



(a) (b)

(c) (d)

Figure 5.2 Four two dimensional synthetic data sets of uncertain data objects, a) SD1, b) SD2, c) SD3 and d) SD4. The correct number of clusters for (a) to (d) are respectively 3,5, 3, and 3.

Figures 5.3-5.6 show the optimal formed clusters for each *k*, *k*=2,3,…,8, after applying the uncertain K-medoids algorithm on SD1-SD4 respectively. As it can be also verified from the figures, the optimal number of clusters should be respectively 3, 5, 3, 3 for SD1-SD4.

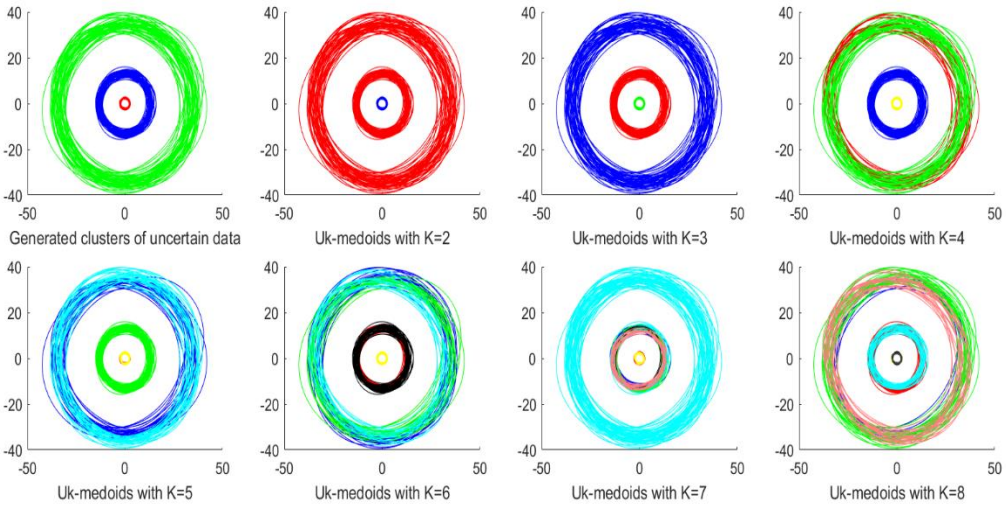


Figure 5.3 The optimal formed clusters for *k*, *k*=2,3,…,8, after applying the uncertain K-medoids algorithm on the two dimensional data set SD1

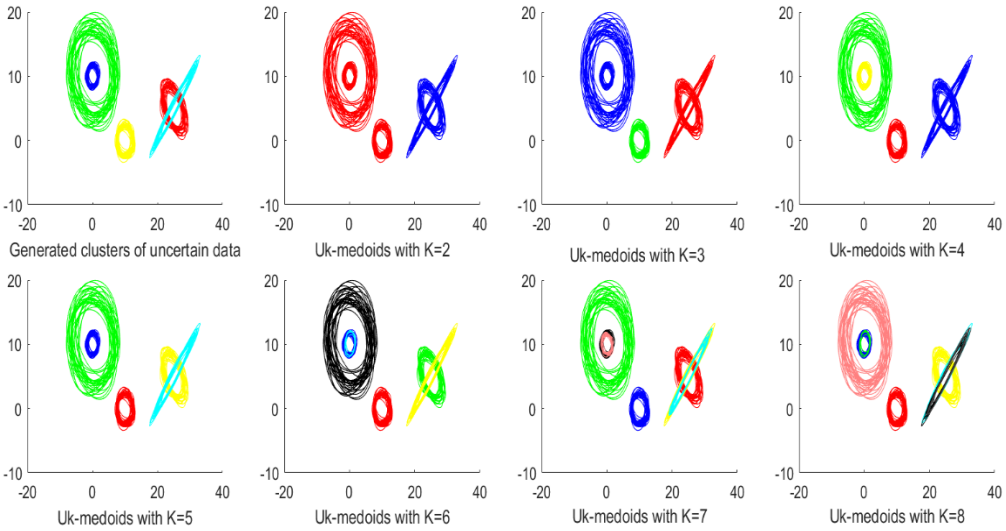


Figure 5.4 The optimal formed clusters for *k*, *k*=2,3,…,8, after applying the uncertain K-medoids algorithm on the two dimensional data set SD2

Tables 5.1-5.3, contain the values of eight different indices: Dunn, Davies-Bouldin, Xie-Beni, Silhouette, uncertain Dunn (OS with r=1, s=t=1), uncertain Silhouette, OS with r=K-1, s=t=3, and OS with r=K-1, s=t=5 for SD1-SD3. Table 5.4 contains one more index compared to Tables 5.1-5.3. That index is OS with r=K-1, s=t=1.

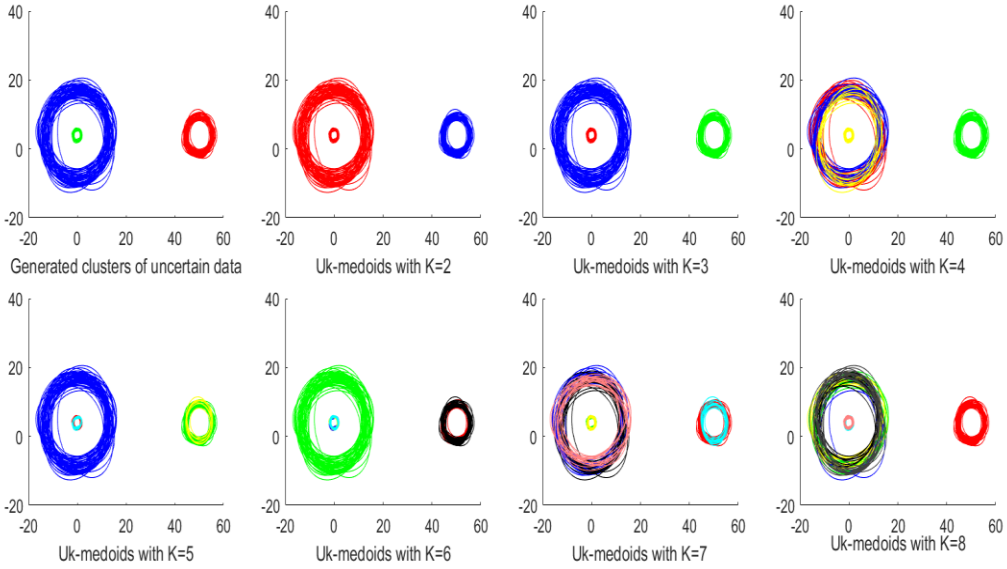


Figure 5.5 The optimal formed clusters for k, k=2,3,…,8, after applying the uncertain K-medoids algorithm on the two dimensional data set SD3

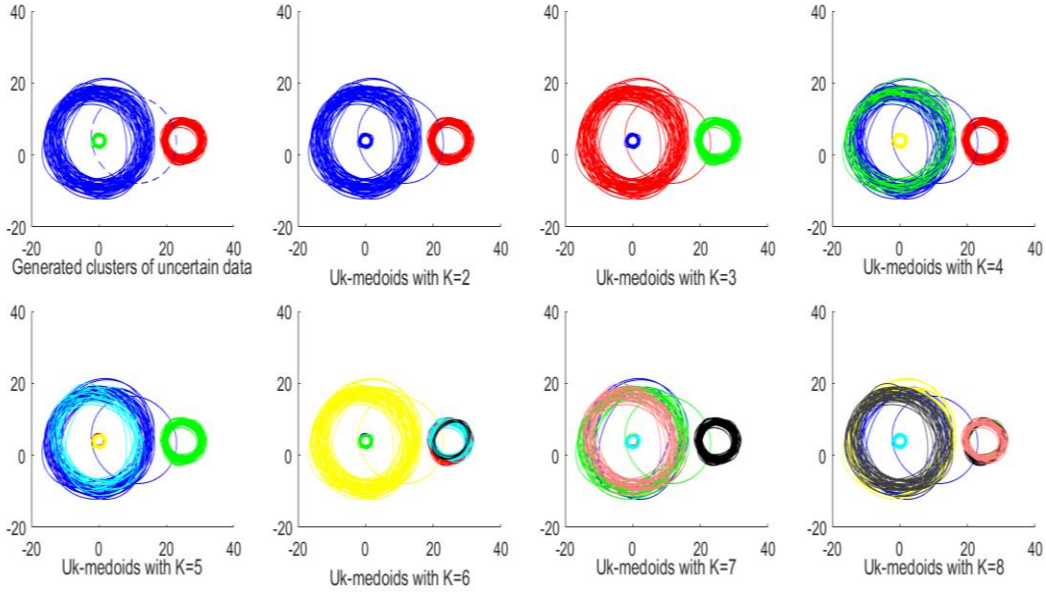


Figure 5.6 The optimal formed clusters for k, k=2,3,…,8, after applying the uncertain K-medoids algorithm on the two dimensional data set SD4

As it can be seen from Table 5.1, Dunn, Davies-Bouldin, Xie-Beni, and Silhouette, the four clustering validity indices for certain data objects that only use the mean of each object, fail in detecting the correct number of clusters for SD1, which is 3. However, it can be seen from the table that the developed clustering validity indices for uncertain data objects: uncertain Dunn (OS with r=1, s=t=1), uncertain Silhouette, OS with r=K-1, s=t=3, and OS with r=K-1, s=t=5 are all successful in detecting the correct number of clusters.

Table 5.1 Applying certain and uncertain clustering validity indices on the two dimensional data set SD1. Uncertain clustering validity indices are all successful in detecting the correct number of clusters while all the certain validity indices fail

K	Dunn	Davies-Bouldin	Xie-Beni	Silhouette	Uncertain Dunn r=1 s=t=1	Uncertain Silhouette	OS r=K-1 s=t=3	OS r=K-1 s=t=5
2	**0.0029**	57.8018	19.0056	-0.1014	1.8400	0.8933	1.7833	1.8400
3	0.0019	20.6739	13.8479	-0.0697	**4.7321**	**0.9817**	**8.1492**	**9.2647**
4	0.0019	16.3579	25.6527	**-0.0634**	0.1479	0.7700	3.4246	4.0633
5	0.0005	28.3027	199.6638	-0.2966	0	0.4666	1.4602	1.7265
6	0.0019	10.1117	**5.3224**	-0.1009	0.1062	0.5019	1.4644	1.7998
7	0.0010	**9.9790**	27.2011	-0.5010	0.0214	0.4158	0.1059	0.1454
8	0.0005	21.1325	838.8978	-0.4815	0	0.2677	0.2240	0.3109

From Table 5.2, it can be seen that in addition to Dunn, Davies-Bouldin, Xie-Beni, and Silhouette, uncertain Dunn (OS with r=1, s=t=1) also fails in detecting the correct number of clusters for SD2, which is 5. The reason for that is large dominant compactness and small dominant separation values. Again, it can be seen from the table that uncertain Silhouette, OS with r=K-1, s=t=3, and OS with r=K-1, s=t=5 are all successful in detecting the correct number of clusters.

Same conclusions are valid for the results of Table 5.3. Again, Dunn, Davies-Bouldin, Xie-Beni, and Silhouette fail because of disability to capture the uncertain nature of the

data objects, and uncertain Dunn (OS with r=1, s=t=1) also fails because of large dominant compactness and small dominant separation values. Uncertain Silhouette, OS with r=K-1, s=t=3, and OS with r=K-1, s=t=5, again successfully detect the correct number of clusters for SD3 which is 3.

Table 5.2 Applying certain and uncertain clustering validity indices on the two dimensional data set SD2. Uncertain clustering validity indices uncertain Silhouette, OS with r=K-1, s=t=3, and OS with r=K-1, s=t=5 are all successful in detecting the correct number of clusters while all the certain validity indices fail

K	Dunn	Davies-Bouldin	Xie-Beni	Silhouette	Uncertain Dunn r=1 s=t=1	Uncertain Silhouette	OS r=K-1 s=t=3	OS r=K-1 s=t=5
2	0.9075	0.3360	0.0006	0.8507	0.2569	0.8285	0.2483	0.2569
3	**4.6451**	**0.0593**	**0.0000**	**0.9975**	**16.2640**	0.9535	44.1609	47.3189
4	0.0092	0.7717	0.0171	0.7411	0.5816	0.9595	44.0676	48.0640
5	0.0092	3.8580	0.4700	0.3686	2.0522	**0.9614**	**57.3715**	**70.1076**
6	0.0092	3.7491	0.4940	0.5140	0.1609	0.8264	19.7402	23.6331
7	0.0018	4.9815	1.2939	0.1897	0	0.6832	1.9928	2.4392
8	0.0100	22.7409	67.6587	0.1742	0.1397	0.6926	2.0621	2.5426

Table 5.3 Applying certain and uncertain clustering validity indices on the two dimensional data set SD3. Uncertain clustering validity indices uncertain Silhouette, OS with r=K-1, s=t=3, and OS with r=K-1, s=t=5 are all successful in detecting the correct number of clusters while all the certain validity indices fail

K	Dunn	Davies-Bouldin	Xie-Beni	Silhouette	Uncertain Dunn r=1 s=t=1	Uncertain Silhouette	OS r=K-1 s=t=3	OS r=K-1 s=t=5
2	**7.7244**	**0.0269**	**0.0000**	**0.9995**	**35.2860**	0.9710	34.3055	35.2860
3	0.0059	12.6706	2.1931	0.4564	4.9203	**0.9899**	**110.6491**	**136.6136**
4	0.0093	7.1799	1.0184	0.5371	0.0173	0.6208	0.0572	0.0710
5	0.0012	11.5813	5.2946	-0.0506	0	0.5502	1.7286	2.0808
6	0.0012	12.1038	19.2728	-0.4720	0	0.4898	1.4815	1.7834
7	0.0011	2.6580	2.7301	-0.0824	0.0759	0.2533	2.3242	2.9036
8	0.0025	6.2771	6.3955	0.2959	0.1178	0.5061	0.2339	0.4163

Table 5.4 results lead to the same conclusion as Table 5.3 except that they also demonstrate the advantage of using OS with r=K-1, s=t=3, and OS with r=K-1, s=t=5, over OS with

$r=K$-1, $s=t=1$ in the case of an existing outlier. As it can be seen from the table, in addition to the validity indices for certain data objects and uncertain Dunn that fail in detecting the correct number of clusters, if OS with $r=K$-1, $s=t=1$ is used, the correct number of clusters which is 3, is also not detected and 2 clusters are detected as the correct number of clusters instead.

Table 5.4 Applying certain and uncertain clustering validity indices on the two dimensional data set SD4. In addition to all the certain validity indices which fail in detecting the correct number of clusters, OS with $r=K$-1, $s=t=1$ also fails due to the existing outlier. Uncertain clustering validity indices uncertain Silhouette, OS with $r=K$-1, $s=t=3$, and OS with $r=K$-1, $s=t=5$ are all successful in detecting the correct number of clusters

K	Dunn	Davies-Bouldin	Xie-Beni	Silhouette	Uncertain Dunn $r=1$ $s=t=1$	Uncertain Silhouette	OS $r=K$-1 $s=t=1$	OS $r=K$-1 $s=t=3$	OS $r=K$-1 $s=t=5$
2	**1.1663**	**0.0721**	**0.0000**	**0.9939**	**8.3014**	0.8868	**7.9472**	8.1401	8.3014
3	0.0085	5.5387	0.4185	0.4156	3.2962	**0.9850**	4.5835	**16.3612**	**25.5424**
4	0.0080	3.0342	0.1601	0.4479	0.0177	0.6729	1.5203	2.9987	4.0152
5	0.0010	3.2160	1.7455	0.1968	0	0.5173	0.5910	1.2890	1.8138
6	0.0010	5.3456	-0.3045	0.0625	0.4267	0.4267	0.5919	1.2668	1.7419
7	0.0010	3.8207	20.9575	0.0819	0	0.4341	0.0613	0.1793	0.2975
8	0.0016	4.3963	3.4262	0.1286	0.0524	0.4476	0.0587	0.1811	0.2991

The values of the studied indices with respect to k, k=2,3,…,8 for SD1-SD4 are shown in Fig. 5.7. As it can be seen from the figures, the developed clustering validity indices for uncertain data uncertain Silhouette, OS with $r=K$-1, $s=t=3$, and OS with $r=K$-1, $s=t=5$ produce relatively sharp peaks for the correct number of clusters.

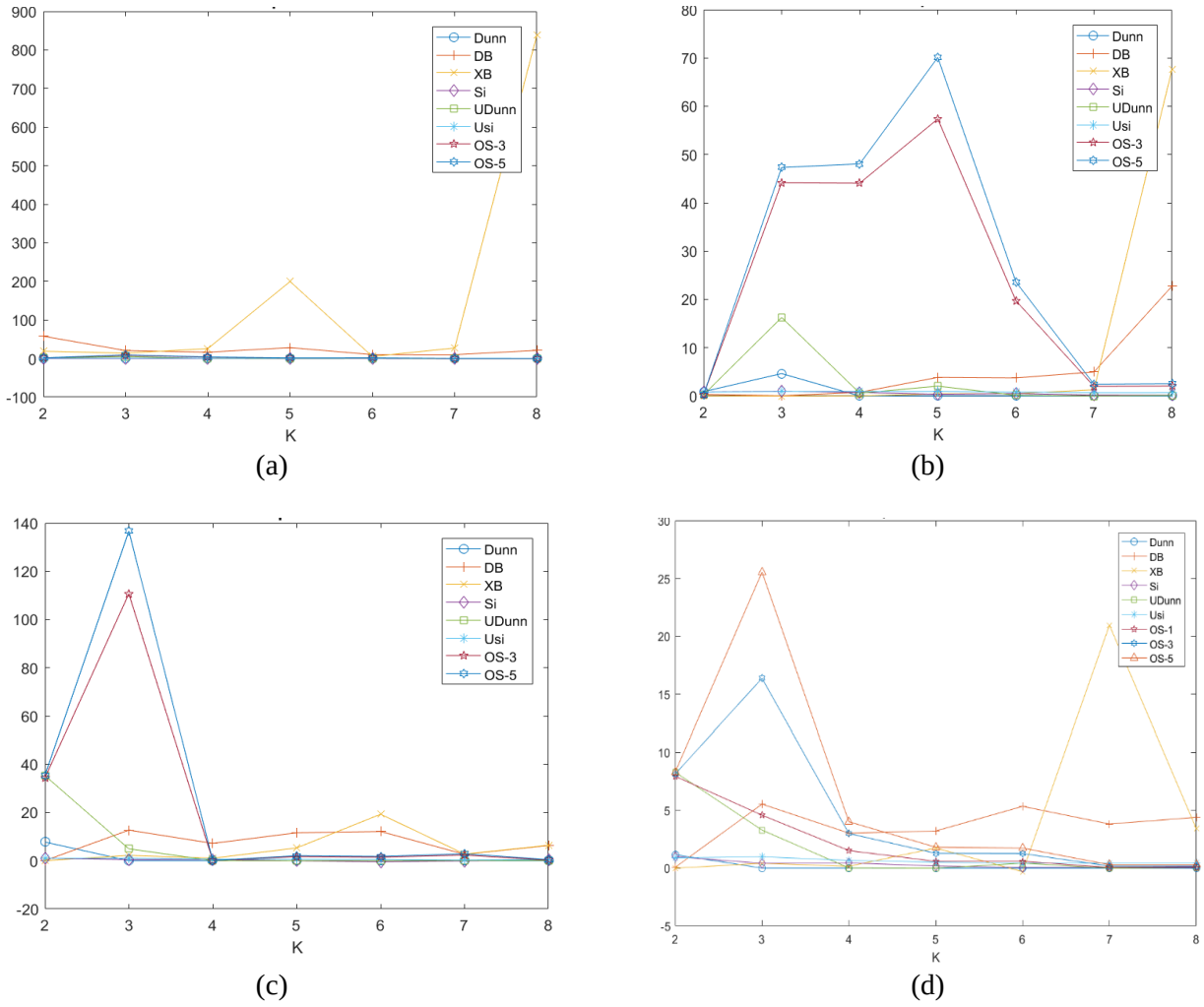


Figure 5.7 Values of the studied indices with respect to *k*, *k*=2,3,…,8, for a) SD1, b) SD2, c) SD3, and d) SD4. The developed clustering validity indices for uncertain data uncertain Silhouette, OS with *r*=*K*-1, *s*=*t*=3, and OS with *r*=*K*-1, *s*=*t*=5 produce relatively sharp peaks for the correct number of clusters.

5.4.2 Three dimensional synthetic data set

In order to show the ability of our proposed indices in detecting the correct number of clusters of uncertain objects with dimensions higher than two, we conducted an experiment on a three-dimensional synthetic data set as well. The data set is called SD5 and contains four generated clusters of uncertain data objects. Fig. 5.8(a) demonstrates three scatterplots for the generated clusters. The scatterplots respectively from left to right, are two-dimensional representations of the uncertain data objects along the first and second

dimensions, the first and third dimensions, and the second and third dimensions. Figures 5.8(b-h), show the optimal formed clusters for each k, k=2,3,…,8, after applying the uncertain K-medoids algorithm on SD5.

For this three-dimensional data set also, as it can be seen from Table 5.5, Dunn, Davies-Bouldin, Xie-Beni, Silhouette, and uncertain Dunn (OS with r=1, s=t=1) fail respectively because of their disability to capture the uncertain nature of the data objects, and large dominant compactness and small dominant separation values. The correct number of clusters for SD5, which is 4, is again successfully detected by uncertain Silhouette, OS with r=K-1, s=t=3, and OS with r=K-1, s=t=5.

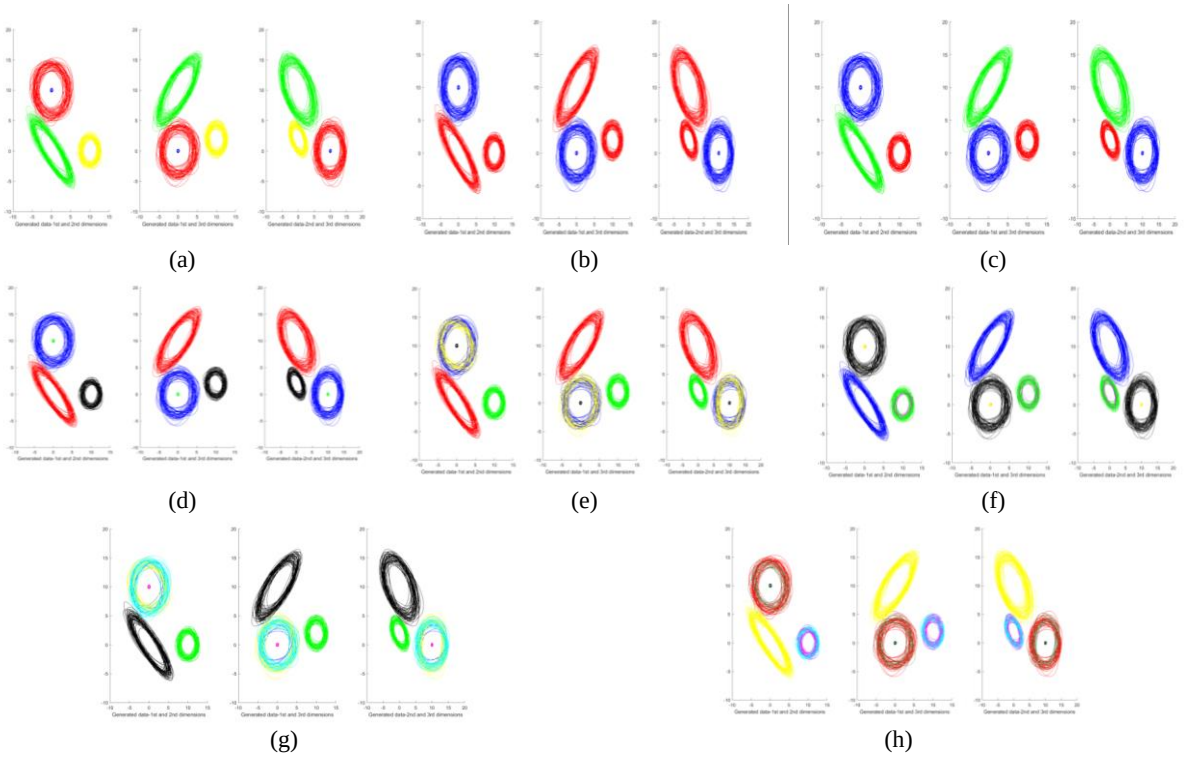


Figure 5.8 Scatterplots of dimensions 1 and 2, 1 and 3, and 2 and 3 for a) the four generated clusters of uncertain data objects, b-h) the optimal formed clusters after applying the uncertain K-medoids algorithm with k, k=2,3,…,8 on the three dimensional data set SD5

Table 5.5 Applying certain and uncertain clustering validity indices on the three-dimensional data set SD5. Uncertain clustering validity indices uncertain Silhouette, OS with $r=K$-1, $s=t=3$, and OS with $r=K$-1, $s=t=5$ are all successful in detecting the correct number of clusters while all the certain validity indices fail

K	Dunn	Davies-Bouldin	Xie-Beni	Silhouette	Uncertain Dunn $r=1$ $s=t=1$	Uncertain Silhouette	OS $r=K$-1 $s=t=3$	OS $r=K$-1 $s=t=5$
2	0.9316	0.5349	0.0013	0.7949	1.6084	0.7613	1.4970	1.6084
3	**4.5841**	**0.0731**	**0.0000**	**0.9978**	**7.2314**	0.9017	22.1590	23.6563
4	0.0151	8.0019	3.0704	0.6033	3.9559	**0.9825**	**46.1158**	**51.1849**
5	0.0151	5.1799	1.3351	0.3406	0.0956	0.7925	19.7584	21.8096
6	0.0028	16.4242	18.2464	0.1221	0.1389	0.5975	17.0608	19.4601
7	0.0028	8.2641	7.3788	0.2546	0.0569	0.6011	2.9664	3.3417
8	0.0030	11.3456	9.9485	0.0173	0	0.3798	1.0924	1.3263

Fig. 5.9 demonstrate the values of the studied indices with respect to k, k=2,3,…,8 for SD5.

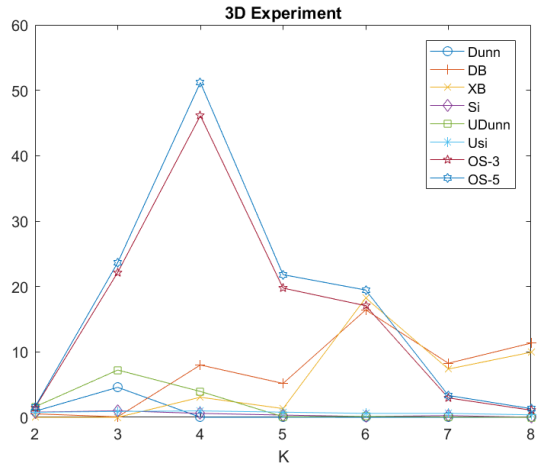
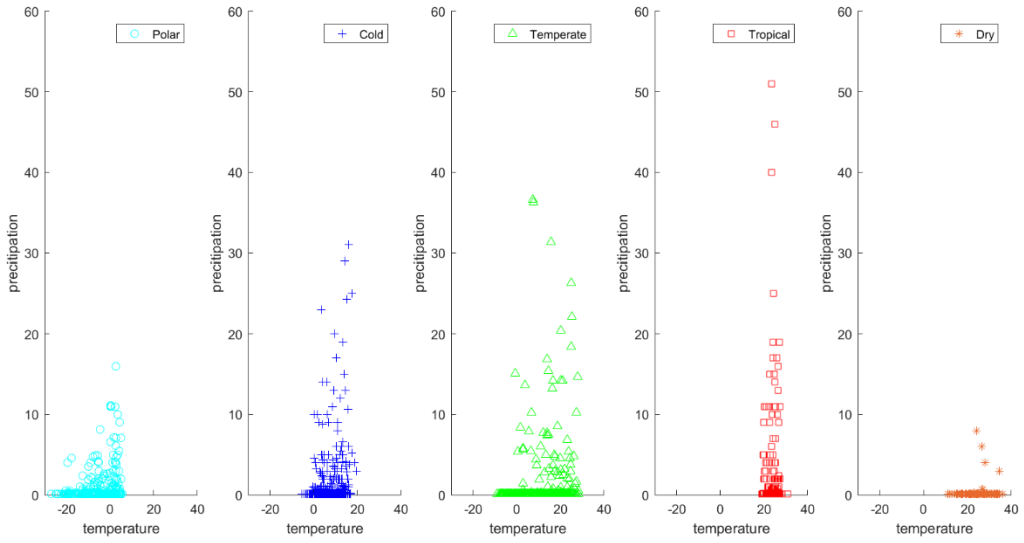


Figure 5.9 Values of the studied indices with respect to k, k=2,3,…,8, for SD5. The developed clustering validity indices for uncertain data uncertain Silhouette, OS with $r=K$-1, $s=t=3$, and OS with $r=K$-1, $s=t=5$ produce relatively sharp peaks for the correct number of clusters.

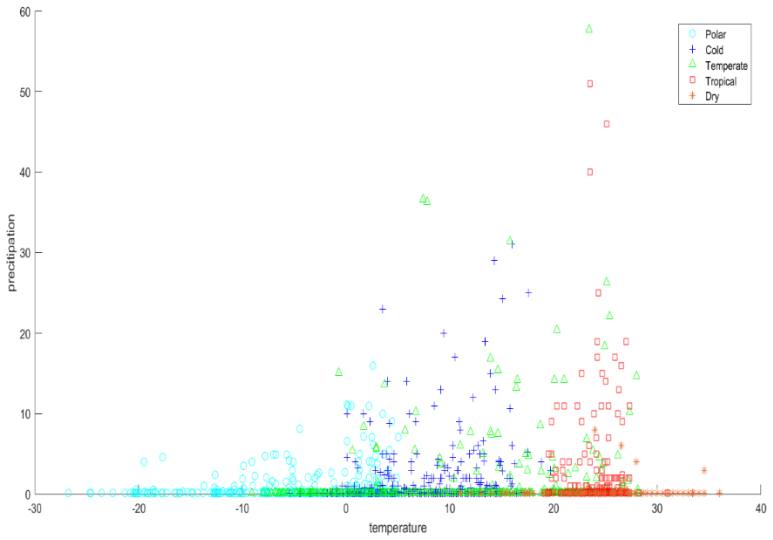
As it can be seen from the figure, uncertain Silhouette, OS with $r=K$-1, $s=t=3$, and OS with $r=K$-1, $s=t=5$ all produce relatively sharp peaks for the correct number of clusters which is four.

5.4.3 Weather data set

The weather data set in this paper is a data set that was collected from the National Center for Atmospheric Research data archive (https://rda.ucar.edu/datasets/ds512.0/). The collected data set contains the daily weather information (average temperature and precipitation level) of 1522 weather stations around the world for the year 2011. Each station in this data set, can be considered as an uncertain object with 365 two-dimensional points. Based on Köppen-Geiger climate classification (Peel et al., 2007), these stations are of five climate types: polar, cold, temperate, tropical, and dry. Fig. 5.10 demonstrates examples of stations with the five climate types.



(a)



(b)

Figure 5.10 Examples of stations with the five climate types: polar, cold, temperate, tropical, dry, a) plotted separately, b) plotted together

We performed the uncertain K-medoids algorithm with Bhattacharyya pdm on the weather data set with k=2,3,…,8. For each k, we ran the algorithm 10 times and compared the performance of eight indices: Dunn, Davies-Bouldin, Xie-Beni, Silhouette, Uncertain Dunn (OS with r=1, s=t=1), Uncertain Silhouette, OS with r=K-1, s=t=3, and OS with r=K-1, s=t=5.The numbers for each particular k in Table 5.6, demonstrate the best results out of the 10 runs for each index.

Table 5.6 Applying certain and uncertain clustering validity indices on the weather data set. Uncertain clustering validity indices uncertain Silhouette, OS with r=K-1, s=t=3, and OS with r=K-1, s=t=5 successfully detect the correct number of clusters which is five while others fail

K	Dunn	Davies-Bouldin	Xie-Beni	Silhouette	Uncertain Dunn r=1 s=t=1	Uncertain Silhouette	OS r=K-1 s=t=3	OS r=K-1 s=t=5
2	0.0002	1.2389	0.0005	**0.4795**	0	0.4466	0	0
3	0.0001	**0.8873**	**0.0004**	0.4387	0.0011	0.4017	0.0017	0.0019
4	0.0001	1.2076	0.0029	0.4899	**0.0404**	0.5197	0.0732	0.0740
5	0.0003	1.0342	0.0025	0.0452	0.0016	**0.5802**	**0.1363**	**0.1375**
6	**0.0006**	1.4505	0.0041	0.0025	0	0.5244	0.1191	0.1198

7	0.0002	2.6987	0.0518	-0.5712	0.0008	0.3363	0.0284	0.0290
8	0.0002	5.3586	0.9361	-0.6211	0.0019	0.3165	0.0651	0.0663

As it can be seen from the table, our developed uncertain clustering validity indices, uncertain Silhouette and OS perform very well in detecting the correct number of clusters (five). The four clustering validity indices for certain data, i.e. Dunn, Davies-Bouldin, Xie-Benie, and Silhouette fail in detecting the correct number of clusters. Also, we can see that Uncertain Dunn, which is a simple case of the OS algorithm, fails, possibly because of its sensitivity to outlier values or either dominant separation values, or compactness values. The values of the eight indices with respect to the number of clusters k, k=2,3,…,8, is plotted in Fig. 5.11.

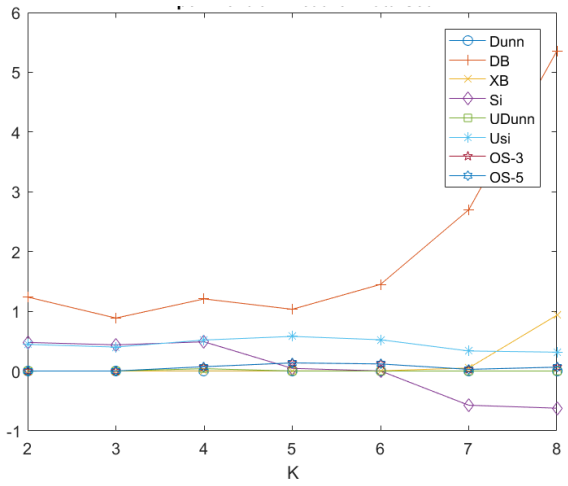


Figure 5.11 Values of the studied indices with respect to k, k=2,3,…,8, for the weather data set. The developed clustering validity indices for uncertain data uncertain Silhouette, OS with r=K-1, s=t=3, and OS with r=K-1, s=t=5 produce relatively sharp peaks for the correct number of clusters.

Table 5.7 shows a summary of the performance of the studied clustering validity indices on all of the data sets. As it can be seen from the table, our proposed clustering validity indices for uncertain data objects, i.e. uncertain Silhouette and OS are both successful in

detecting the correct number of clusters for all of the data sets. Uncertain Dunn which is a reduced and simplified form of the OS is only successful for one data set (SD1), while all the clustering validity indices for certain data objects, i.e. Dunn, Davies-Bouldin, Xie-Beni, and Silhouette, fail to detect the correct number of clusters for all data sets.

Table 5.7 Summary of the performance of the studied clustering validity indices on all of the data sets. The developed uncertain clustering validity indices successfully detect the correct number of clusters for all the studied data sets while uncertain Dunn is only successful for one data set and the certain clustering validity indices fail for all of the data sets

Data set	True clusters	Dunn	Davies-Bouldin	Xie-Beni	Silhouette	Uncertain Dunn $r=1$ $s=t=1$	Uncertain Silhouette	OS $r=K-1$ $s=t=3$	OS $r=K-1$ $s=t=5$
SD1	3	1	7	6	4	**3**	**3**	**3**	**3**
SD2	5	3	3	3	3	3	**5**	**5**	**5**
SD3	3	2	2	2	2	2	**3**	**3**	**3**
SD4	3	2	2	2	2	2	**3**	**3**	**3**
SD5	4	3	3	3	3	3	**4**	**4**	**4**
Weather	5	6	3	3	2	4	**5**	**5**	**5**

Fig. 5.12, demonstrates the values of the studied indices: Dunn, Davies-Bouldin, Xie-Beni, Silhouette, uncertain Dunn, uncertain Silhouette, OS with $r=K-1$, $s=t=3$, and OS with $r=K-1$, $s=t=5$, with respect to k, $k=2,3,\ldots,8$ for all of the data sets: SD1, SD2, SD3, SD4, SD5, and weather. As it can be seen from the figure, uncertain Silhouette, OS with $r=K-1$, $s=t=3$, and OS with $r=K-1$, $s=t=5$ are the only indices that produce sharp peaks for the correct number of clusters of all of the studied data sets.

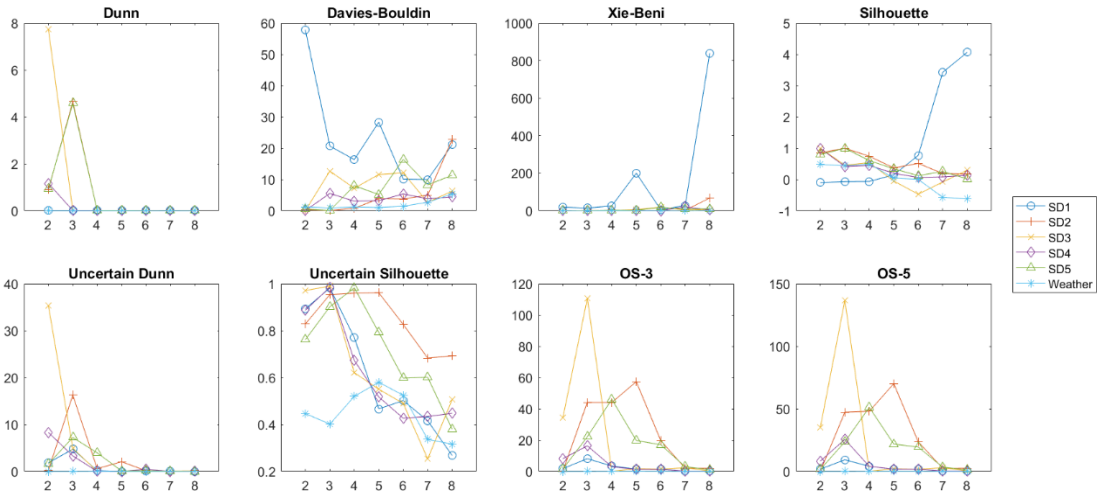


Figure 5.12 Values of the studied indices with respect to k, k=2,3,…,8, for all of the data sets. Uncertain Silhouette, OS with r=K-1, s=t=3, and OS with r=K-1, s=t=5 are the only indices that produce sharp peaks for the correct number of clusters of all of the studied data sets

5.5 Conclusion

In this chapter, we proposed two indices, named uncertain Silhouette and Order Statistics index (OS), for validation of clusters of uncertain data objects. To our best knowledge, prior to this work, there was not any clustering validity indices in the literature, designed to handle uncertain objects given in forms of multiple points or probability density functions.

Our proposed indices not only outperform existing certain clustering validity indices in validating clusters of uncertain data objects, they also show robustness to existence of outlier objects. We particularly designed the OS index to handle the type of problems where either a very scattered cluster or two very close clusters, can impede detection of correct number of clusters. We showed in the experiments that the OS index clearly outperforms uncertain Dunn (also developed here) which is a simplified case of the OS index in dealing

with such cases. The effectiveness of our developed indices was evaluated through several experiments on synthetic and real data sets.

Besides the two developed clustering validity indices in this dissertation, more indices for uncertain data objects can be significant in a conducting a comprehensive validation of clusters of uncertain data objects.

CHAPTER 6 Conclusion

In this dissertation, we proposed two new approaches for classifying uncertain objects modeled with multivariate Normal PDF. Both of the proposed approaches are based on the concept of probabilistic distance measures. The first approach is based on obtaining object-to-object distances. It includes a K-nearest neighbor classifier that can use existing probabilistic distance measures. We choose Bhattacharyya distance measure as the probabilistic distance measure since it has analytical solution for multivariate Normal PDF. This approach is successful in classifying uncertain objects in experiment using both simulated and real data as it proves to be better than the certain KNN approach and uncertain naïve Bayesian classifier in majority of the verified cases.

In order to achieve even better classification performance another approach was proposed. The second approach is based on the object-to-group distance. In this regard, it uses a proposed probabilistic distance measure called OGPDM. Using OGPDM for classification both object-correlation and class-correlation are captured. The OGPDM approach provides better classification performance compared to the other approaches as it uses the optimal separating hyper-plane.

We also defined measures of scatter for uncertain data objects. We developed the definition of covariance matrix for uncertain data objects. Within and between scatter matrices were also defined for uncertain objects. Using the developed measures of scatter, we extended the Fisher Linear Discriminant Analysis for uncertain data objects. Also we developed Kernel Fisher Discriminants for uncertain data objects. The derivations were developed for two cases: 1) when uncertain objects are given with probability density functions, 2) when uncertain objects are given with multiple points. We showed through

examples that the obtained decision boundaries from our developed Fisher Discriminants for uncertain objects seem very reasonable for separating classes of uncertain objects. Moreover, we showed scenarios with uncertain objects modeled with skew-normal distribution that when the distance between classes of the uncertain objects is small our developed Uncertain Fisher Discriminants performs better than Certain Fisher Discriminants in terms of classification accuracy.

Finally, we proposed two indices, named uncertain Silhouette and Order Statistics index (OS), for validation of clusters of uncertain data objects. To our best knowledge, prior to this work, there was not any clustering validity indices in the literature, designed to handle uncertain objects given in forms of multiple points or probability density functions.

Our proposed indices not only outperform existing certain clustering validity indices in validating clusters of uncertain data objects, they also show robustness to existence of outlier objects. We particularly designed the OS index to handle the type of problems where either a very scattered cluster or two very close clusters, can impede detection of correct number of clusters. We showed in the experiments that the OS index clearly outperforms uncertain Dunn (also developed here) which is a simplified case of the OS index in dealing with such cases. The effectiveness of our developed indices was evaluated through several experiments on synthetic and real data sets.

Appendix I.a: Kernelizing $\mathbf{w^t B^U w}$ for objects given with multiple points

$$\mathbf{w}^{\mathbf{t}}\mathrm{B}_{\emptyset}^{\mathrm{U}}\mathbf{w} = \left[\sum_{\mathrm{i}=1}^{\mathrm{n}} \alpha_{\mathrm{i}}\emptyset(\widehat{\boldsymbol{\mu}}_{\mathbf{i}})\right]^{\mathbf{t}} \left[(\widehat{\boldsymbol{\mu}}_{\mathbf{1}}^{\emptyset} - \widehat{\boldsymbol{\mu}}_{\mathbf{2}}^{\emptyset})(\widehat{\boldsymbol{\mu}}_{\mathbf{1}}^{\emptyset} - \widehat{\boldsymbol{\mu}}_{\mathbf{2}}^{\emptyset})^{\mathrm{t}}\right] \left[\sum_{\mathrm{i}=1}^{\mathrm{n}} \alpha_{\mathrm{i}}\emptyset(\widehat{\boldsymbol{\mu}}_{\mathbf{i}})\right]$$

$$= \left[\sum_{\mathbf{i}=\mathbf{1}}^{\mathbf{n}} \boldsymbol{\alpha}_{\mathbf{i}}\emptyset(\widehat{\boldsymbol{\mu}}_{\mathbf{i}})\right]^{\mathrm{t}} \left[\left(\frac{1}{\mathrm{n}_1}\sum_{\mathbf{j}=\mathbf{1}}^{\mathbf{n_1}} \emptyset\left(\widehat{\boldsymbol{\mu}}_{\mathbf{j}}^{\mathbf{1}}\right)\right.\right.$$

$$\left.\left. - \frac{1}{\mathrm{n}_2}\sum_{\mathbf{j}=\mathbf{1}}^{\mathbf{n_2}} \emptyset\left(\widehat{\boldsymbol{\mu}}_{\mathbf{j}}^{\mathbf{2}}\right)\right)\left(\frac{1}{\mathrm{n}_1}\sum_{\mathbf{j}=\mathbf{1}}^{\mathbf{n_1}} \emptyset\left(\widehat{\boldsymbol{\mu}}_{\mathbf{j}}^{\mathbf{1}}\right) - \frac{1}{\mathrm{n}_2}\sum_{\mathbf{j}=\mathbf{1}}^{\mathbf{n_2}} \emptyset\left(\widehat{\boldsymbol{\mu}}_{\mathbf{j}}^{\mathbf{2}}\right)\right)^{\mathrm{t}}\right] \left[\sum_{\mathbf{i}=\mathbf{1}}^{\mathbf{n}} \boldsymbol{\alpha}_{\mathbf{i}}\emptyset(\widehat{\boldsymbol{\mu}}_{\mathbf{i}})\right]$$

$$= \left[\left(\frac{1}{\mathrm{n}_1}\sum_{\mathrm{i}=1}^{\mathrm{n}}\sum_{\mathrm{j}=1}^{\mathrm{n}_1} \alpha_{\mathrm{i}}\emptyset(\widehat{\boldsymbol{\mu}}_{\mathbf{i}})^{\mathrm{t}}\emptyset\left(\widehat{\boldsymbol{\mu}}_{\mathbf{j}}^{\mathbf{1}}\right)\right)\right.$$

$$\left. - \left(\frac{1}{\mathrm{n}_2}\sum_{\mathrm{i}=1}^{\mathrm{n}}\sum_{\mathrm{j}=1}^{\mathrm{n}_2} \alpha_{\mathrm{i}}\emptyset(\widehat{\boldsymbol{\mu}}_{\mathbf{i}})^{\mathrm{t}}\emptyset\left(\widehat{\boldsymbol{\mu}}_{\mathbf{j}}^{\mathbf{2}}\right)\right)\right]\left[\left(\frac{1}{\mathrm{n}_1}\sum_{\mathrm{i}=1}^{\mathrm{n}}\sum_{\mathrm{j}=1}^{\mathrm{n}_1} \emptyset\left(\widehat{\boldsymbol{\mu}}_{\mathbf{j}}^{\mathbf{1}}\right)^{\mathrm{t}}\emptyset(\widehat{\boldsymbol{\mu}}_{\mathbf{i}})\alpha_{\mathrm{i}}\right)\right.$$

$$\left. - \left(\frac{1}{\mathrm{n}_2}\sum_{\mathrm{i}=1}^{\mathrm{n}}\sum_{\mathrm{j}=1}^{\mathrm{n}_2} \emptyset\left(\widehat{\boldsymbol{\mu}}_{\mathbf{j}}^{\mathbf{2}}\right)^{\mathbf{t}}\emptyset(\widehat{\boldsymbol{\mu}}_{\mathbf{i}})\alpha_{\mathrm{i}}\right)\right] =$$

$$\left(\boldsymbol{\alpha}^{\mathbf{t}}\begin{bmatrix}\frac{1}{n_1}\sum_{j=1}^{n_1}k\left(\widehat{\boldsymbol{\mu}}_{\mathbf{1}},\widehat{\boldsymbol{\mu}}_{\mathbf{j}}^{\mathbf{1}}\right)\\ \vdots\\ \frac{1}{n_1}\sum_{j=1}^{n_1}k\left(\widehat{\boldsymbol{\mu}}_{\mathbf{n}},\widehat{\boldsymbol{\mu}}_{\mathbf{j}}^{\mathbf{1}}\right)\end{bmatrix}-\boldsymbol{\alpha}^{\mathbf{t}}\begin{bmatrix}\frac{1}{n_2}\sum_{j=1}^{n_2}k\left(\widehat{\boldsymbol{\mu}}_{\mathbf{1}},\widehat{\boldsymbol{\mu}}_{\mathbf{j}}^{\mathbf{2}}\right)\\ \vdots\\ \frac{1}{n_2}\sum_{j=1}^{n_2}k\left(\widehat{\boldsymbol{\mu}}_{\mathbf{n}},\widehat{\boldsymbol{\mu}}_{\mathbf{j}}^{\mathbf{2}}\right)\end{bmatrix}\right)\left(\begin{bmatrix}\frac{1}{n_1}\sum_{j=1}^{n_1}k\left(\widehat{\boldsymbol{\mu}}_{\mathbf{1}},\widehat{\boldsymbol{\mu}}_{\mathbf{j}}^{\mathbf{1}}\right)\\ \vdots\\ \frac{1}{n_1}\sum_{j=1}^{n_1}k\left(\widehat{\boldsymbol{\mu}}_{\mathbf{n}},\widehat{\boldsymbol{\mu}}_{\mathbf{j}}^{\mathbf{1}}\right)\end{bmatrix}^{\mathbf{t}}\boldsymbol{\alpha}\right.$$

$$\left.-\begin{bmatrix}\frac{1}{n_2}\sum_{j=1}^{n_2}k\left(\widehat{\boldsymbol{\mu}}_{\mathbf{1}},\widehat{\boldsymbol{\mu}}_{\mathbf{j}}^{\mathbf{2}}\right)\\ \vdots\\ \frac{1}{n_2}\sum_{j=1}^{n_2}k\left(\widehat{\boldsymbol{\mu}}_{\mathbf{n}},\widehat{\boldsymbol{\mu}}_{\mathbf{j}}^{\mathbf{2}}\right)\end{bmatrix}^{t}\boldsymbol{\alpha}\right)=\boldsymbol{\alpha}^{\mathbf{t}}\left(\mathbf{m_{p1}}-\mathbf{m_{p2}}\right)\left(\mathbf{m_{p1}}-\mathbf{m_{p2}}\right)^{\mathbf{t}}\boldsymbol{\alpha}=\boldsymbol{\alpha}^{\mathbf{t}}M_p\boldsymbol{\alpha}$$

where $\boldsymbol{\alpha}^{\mathbf{t}}=[\alpha_1,\alpha_2,\dots,\alpha_n]$; $M_p=\left(\mathbf{m_{p1}}-\mathbf{m_{p2}}\right)\left(\mathbf{m_{p1}}-\mathbf{m_{p2}}\right)^{\mathbf{t}}$; $\mathbf{m_{pk}}=$

$$\begin{bmatrix}\frac{1}{n_k}\sum_{j=1}^{n_k}k\left(\widehat{\boldsymbol{\mu}}_{\mathbf{1}},\widehat{\boldsymbol{\mu}}_{\mathbf{j}}^{\mathbf{k}}\right)\\ \vdots\\ \frac{1}{n_k}\sum_{j=1}^{n_k}k\left(\widehat{\boldsymbol{\mu}}_{\mathbf{n}},\widehat{\boldsymbol{\mu}}_{\mathbf{j}}^{\mathbf{k}}\right)\end{bmatrix}.$$

Appendix I.b: Kernelizing $\mathbf{w^tW^Uw}$ for objects given with multiple points

$\mathbf{w}^{t}W_U^{\emptyset}\mathbf{w}=$

$$\left[\sum_{i=1}^{n}\alpha_i\emptyset\left(\widehat{\boldsymbol{\mu}}_{\mathbf{i}}\right)\right]^{\mathbf{t}}\left[\sum_{k=1}^{K}\sum_{j=1}^{n_k}\left[\left(\emptyset\left(\widehat{\boldsymbol{\mu}}_{\mathbf{j}}^{\mathbf{k}}\right)-\widehat{\boldsymbol{\mu}}_{\mathbf{k}}^{\emptyset}\right)\left(\emptyset\left(\widehat{\boldsymbol{\mu}}_{\mathbf{j}}^{\mathbf{k}}\right)-\widehat{\boldsymbol{\mu}}_{\mathbf{k}}^{\emptyset}\right)^{\mathbf{t}}\right.\right.$$

$$\left.\left.+\frac{\sum_{t}^{l_j^k}\left(\emptyset\left(\mathbf{x}_{\mathbf{tj}}^{\mathbf{k}}\right)-\emptyset\left(\widehat{\boldsymbol{\mu}}_{\mathbf{j}}^{\mathbf{k}}\right)\right)\left(\emptyset\left(\mathbf{x}_{\mathbf{tj}}^{\mathbf{k}}\right)-\emptyset\left(\widehat{\boldsymbol{\mu}}_{\mathbf{j}}^{\mathbf{k}}\right)\right)^{\mathbf{t}}}{l_j^k}\right]\right]\left[\sum_{i=1}^{n}\alpha_i\emptyset\left(\widehat{\boldsymbol{\mu}}_{\mathbf{i}}\right)\right]$$

where it can be further simplified into a two-parts formula:

$$\mathbf{w}^{t} W_{U}^{\emptyset} \mathbf{w}$$

$$= \left[\sum_{i=1}^{n} \alpha_i \emptyset(\widehat{\boldsymbol{\mu}}_{\mathbf{i}})\right]^{\mathbf{t}} \left[\sum_{k=1}^{K} \sum_{j=1}^{n_k} \left[\left(\emptyset\left(\widehat{\boldsymbol{\mu}}_{\mathbf{j}}^{\mathbf{k}}\right) - \widehat{\boldsymbol{\mu}}_{\mathbf{k}}^{\emptyset}\right)\left(\emptyset\left(\widehat{\boldsymbol{\mu}}_{\mathbf{j}}^{\mathbf{k}}\right) - \widehat{\boldsymbol{\mu}}_{\mathbf{k}}^{\emptyset}\right)^{\mathbf{t}}\right]\right] \left[\sum_{i=1}^{n} \alpha_i \emptyset(\widehat{\boldsymbol{\mu}}_{\mathbf{i}})\right]$$

$$+ \left[\sum_{i=1}^{n} \alpha_i \emptyset(\widehat{\boldsymbol{\mu}}_{\mathbf{i}})\right]^{\mathbf{t}} \left[\sum_{k=1}^{K} \sum_{j=1}^{n_k} \left[\frac{\sum_{t}^{l_j^k} \left(\emptyset\left(\mathbf{x}_{\mathbf{tj}}^{\mathbf{k}}\right) - \emptyset\left(\widehat{\boldsymbol{\mu}}_{\mathbf{j}}^{\mathbf{k}}\right)\right)\left(\emptyset\left(\mathbf{x}_{\mathbf{tj}}^{\mathbf{k}}\right) - \emptyset\left(\widehat{\boldsymbol{\mu}}_{\mathbf{j}}^{\mathbf{k}}\right)\right)^{\mathbf{t}}}{l_j^k}\right]\right] \left[\sum_{i=1}^{n} \alpha_i \emptyset(\widehat{\boldsymbol{\mu}}_{\mathbf{i}})\right]$$

Same as the derived formula by (S. Mika, et al., 1999), Part 1 of the above formula can be written as:

$$\left[\sum_{i=1}^{n} \alpha_i \emptyset(\widehat{\boldsymbol{\mu}}_{\mathbf{i}})\right]^{\mathbf{t}} \left[\sum_{k=1}^{K} \sum_{j=1}^{n_k} \left[\left(\emptyset\left(\widehat{\boldsymbol{\mu}}_{\mathbf{j}}^{\mathbf{k}}\right) - \widehat{\boldsymbol{\mu}}_{\mathbf{k}}^{\emptyset}\right)\left(\emptyset\left(\widehat{\boldsymbol{\mu}}_{\mathbf{j}}^{\mathbf{k}}\right) - \widehat{\boldsymbol{\mu}}_{\mathbf{k}}^{\emptyset}\right)^{t}\right]\right] \left[\sum_{i=1}^{n} \alpha_i \emptyset(\widehat{\boldsymbol{\mu}}_{\mathbf{i}})\right]$$

$$= \boldsymbol{\alpha}^{\mathbf{t}} \sum_{k=1}^{K} U_{p1k}(I - 1_{n_k})\, U_{p1k}{}^{t} \boldsymbol{\alpha} = \boldsymbol{\alpha}^{\mathbf{t}} N_{p1} \boldsymbol{\alpha}$$

where

$$\boldsymbol{\alpha}^{\mathbf{t}} = [\alpha_1, \alpha_2, \ldots, \alpha_n]; \quad U_{p1k} = \begin{bmatrix} k\left(\widehat{\boldsymbol{\mu}}_{\mathbf{1}}, \widehat{\boldsymbol{\mu}}_{\mathbf{1}}^{\mathbf{k}}\right) & \cdots & k\left(\widehat{\boldsymbol{\mu}}_{\mathbf{1}}, \widehat{\boldsymbol{\mu}}_{\mathbf{n_k}}^{\mathbf{k}}\right) \\ \vdots & \ddots & \vdots \\ k\left(\widehat{\boldsymbol{\mu}}_{\mathbf{n}}, \widehat{\boldsymbol{\mu}}_{\mathbf{1}}^{\mathbf{k}}\right) & \cdots & k\left(\widehat{\boldsymbol{\mu}}_{\mathbf{n}}, \widehat{\boldsymbol{\mu}}_{\mathbf{n_k}}^{\mathbf{k}}\right) \end{bmatrix}; \; 1_{n_k} = \begin{bmatrix} 1/n_k & \cdots & 1/n_k \\ \vdots & \ddots & \vdots \\ 1/n_k & \cdots & 1/n_k \end{bmatrix}$$

Part 2 of the formula can be expanded as follow:

$$\left[\sum_{i=1}^{n} \alpha_i \emptyset(\widehat{\boldsymbol{\mu}}_{\mathbf{i}})\right]^{\mathbf{t}} \left[\sum_{k=1}^{K} \sum_{j=1}^{n_k} \left[\frac{\sum_{t}^{l_j^k} \left(\emptyset\left(\mathbf{x}_{\mathbf{tj}}^{\mathbf{k}}\right) - \emptyset\left(\widehat{\boldsymbol{\mu}}_{\mathbf{j}}^{\mathbf{k}}\right)\right)\left(\emptyset\left(\mathbf{x}_{\mathbf{tj}}^{\mathbf{k}}\right) - \emptyset\left(\widehat{\boldsymbol{\mu}}_{\mathbf{j}}^{\mathbf{k}}\right)\right)^{t}}{l_j^k}\right]\right] \left[\sum_{i=1}^{n} \alpha_i \emptyset(\widehat{\boldsymbol{\mu}}_{\mathbf{i}})\right]$$

$$= \sum_{i=1}^{n} \sum_{k=1}^{K} \sum_{j=1}^{n_k} \left[\frac{\sum_{t}^{l_j^k} \alpha_i \emptyset(\widehat{\boldsymbol{\mu}}_{\mathbf{i}})^{\mathbf{t}} \left(\emptyset\left(\mathbf{x}_{\mathbf{tj}}^{\mathbf{k}}\right) - \emptyset\left(\widehat{\boldsymbol{\mu}}_{\mathbf{j}}^{\mathbf{k}}\right)\right)\left(\emptyset\left(\mathbf{x}_{\mathbf{tj}}^{\mathbf{k}}\right) - \emptyset\left(\widehat{\boldsymbol{\mu}}_{\mathbf{j}}^{\mathbf{k}}\right)\right)^{t} \emptyset(\widehat{\boldsymbol{\mu}}_{\mathbf{i}})\, \alpha_i}{l_j^k}\right] =$$

$$\sum_{k=1}^{K}\sum_{i=1}^{n}\sum_{j=1}^{n_k}\left[\frac{\sum_{t}^{l_j^k}\alpha_i\emptyset(\hat{\boldsymbol{\mu}}_i)'\left(\emptyset\left(\mathbf{x_{tj}^k}\right)-\frac{1}{l_j^k}\sum_{t=1}^{l_j^k}\emptyset(\mathbf{x_{tj}^k})\right)\left(\emptyset\left(\mathbf{x_{tj}^k}\right)-\frac{1}{l_j^k}\sum_{t=1}^{l_j^k}\emptyset(\mathbf{x_{tj}^k})\right)^{t}\emptyset(\hat{\boldsymbol{\mu}}_i)\alpha_i}{l_j^k}\right]$$

$$=\boldsymbol{\alpha^t}\sum_{k=1}^{K}U_{p2k}(I-1_{l_j^k})\,U_{p2k}^{t}\boldsymbol{\alpha}=\boldsymbol{\alpha^t}N_{p2}\boldsymbol{\alpha}$$

where $\boldsymbol{\alpha^t}=[\alpha_1,\alpha_2,\dots,\alpha_n]$; $U_{p2k}=\begin{bmatrix}\sum_{j=1}^{n_k}k\left(\hat{\boldsymbol{\mu}}_{\mathbf{1}},\mathbf{x_{1j}^k}\right) & \cdots & \sum_{j=1}^{n_k}k\left(\hat{\boldsymbol{\mu}}_{\mathbf{1}},\mathbf{x_{l_j^k j}^k}\right)\\ \vdots & \ddots & \vdots\\ \sum_{j=1}^{n_k}k\left(\hat{\boldsymbol{\mu}}_{\mathbf{n}},\mathbf{x_{1j}^k}\right) & \cdots & \sum_{j=1}^{n_k}k\left(\hat{\boldsymbol{\mu}}_{\mathbf{n}},\mathbf{x_{l_j^k j}^k}\right)\end{bmatrix}$;

$$1_{l_j^k}=\begin{bmatrix}1/l_j^k & \cdots & 1/l_j^k\\ \vdots & \ddots & \vdots\\ 1/l_j^k & \cdots & 1/l_j^k\end{bmatrix}$$

Appendix II: Kernelizing $\mathbf{w^t W^U w}$ for objects given with PDF

$$\mathbf{w^t} W_U^{\emptyset}\mathbf{w} = \left[\sum_{i=1}^{n} \alpha_i \emptyset(\boldsymbol{\mu_i})\right]^t \left[\sum_{k=1}^{K}\sum_{j=1}^{n_k}\left[\left(\emptyset\left(\boldsymbol{\mu_j^k}\right) - \boldsymbol{\mu_k^{\emptyset}}\right)\left(\emptyset\left(\boldsymbol{\mu_j^k}\right) - \boldsymbol{\mu_k^{\emptyset}}\right)^t\right]\right]\left[\sum_{i=1}^{n} \alpha_i \emptyset(\boldsymbol{\mu_i})\right]$$

$$+ \left[\sum_{i=1}^{n} \alpha_i \emptyset(\boldsymbol{\mu_i})\right]^t \left[\sum_{k=1}^{K}\sum_{j=1}^{n_k}\left[(\Sigma_j^k)^{\emptyset}\right]\right]\left[\sum_{i=1}^{n} \alpha_i \emptyset(\boldsymbol{\mu_i})\right]$$

Again, the first part of the formulation above is the same as the one developed for Uncertain Kernel Fisher Discriminant for objects given with multiple points. Instead of using estimates for mean vectors, actual parameter values are used. Therefore, we can write it as:

$$\left[\sum_{i=1}^{n} \alpha_i \emptyset(\boldsymbol{\mu_i})\right]^{\mathbf{t}} \left[\sum_{k=1}^{K}\sum_{j=1}^{n_k}\left[\left(\emptyset\left(\boldsymbol{\mu_j^k}\right) - \boldsymbol{\mu_k^{\emptyset}}\right)\left(\emptyset\left(\boldsymbol{\mu_j^k}\right) - \boldsymbol{\mu_k^{\emptyset}}\right)^{\mathbf{t}}\right]\right]\left[\sum_{i=1}^{n} \alpha_i \emptyset(\boldsymbol{\mu_i})\right] = \boldsymbol{\alpha^t} N_{d1} \boldsymbol{\alpha}$$

where $N_{d1} = \sum_{k=1}^{K} U_{d1k}(I - 1_{n_k})\, U_{d1k}{}^t$; $\boldsymbol{\alpha^t} = [\alpha_1, \alpha_2, \dots, \alpha_n]$;

$$U_{d1k} = \begin{bmatrix} k\left(\boldsymbol{\mu_1}, \boldsymbol{\mu_1^k}\right) & \cdots & k\left(\boldsymbol{\mu_1}, \boldsymbol{\mu_{n_k}^k}\right) \\ \vdots & \ddots & \vdots \\ k\left(\boldsymbol{\mu_n}, \boldsymbol{\mu_1^k}\right) & \cdots & k\left(\boldsymbol{\mu_n}, \boldsymbol{\mu_{n_k}^k}\right) \end{bmatrix};\ 1_{n_k} = \begin{bmatrix} 1/n_k & \cdots & 1/n_k \\ \vdots & \ddots & \vdots \\ 1/n_k & \cdots & 1/n_k \end{bmatrix}.$$

The second part of the formulation includes the mapped object-covariance matrices. Using the idea in (J.Yang & S.Gunn, 2007), the mapped covariance matrix of object j in class k can be written as: $(\Sigma_j^k)^{\emptyset} = J_j^k \Sigma_j^k (J_j^k)^T$ where $J_j^k = \frac{\partial \emptyset(\mathbf{X})}{\partial \boldsymbol{\mu_j^k}}$ is the Jacobian matrix for the mapped vector examined at $\boldsymbol{\mu_j^k}$ (the mean of object j in class k).

The second part of the formulation can be expanded as follows:

$$\left[\sum_{i=1}^{n}\alpha_i\emptyset(\boldsymbol{\mu_i})\right]^{\mathbf{t}}\left[\sum_{k=1}^{K}\sum_{j=1}^{n_k}\left[(\sum_j^k)^{\emptyset}\right]\right]\left[\sum_{i=1}^{n}\alpha_i\emptyset(\boldsymbol{\mu_i})\right]$$

$$=\left[\sum_{\mathbf{i=1}}^{\mathbf{n}}\boldsymbol{\alpha_i}\emptyset(\boldsymbol{\mu_i})\right]^{\mathbf{t}}\left[\sum_{\mathbf{k=1}}^{\mathbf{K}}\sum_{\mathbf{j=1}}^{\mathbf{n_k}}J_j^k\sum_j^k(J_j^k)^T\right]\left[\sum_{\mathbf{i=1}}^{\mathbf{n}}\boldsymbol{\alpha_i}\emptyset(\boldsymbol{\mu_i})\right]$$

As $\sum_j^k$ is a real symmetric matrix, it can be decomposed by its eigenvalues and eigenvectors $\sum_j^k = Q_j^k\Lambda_j^kQ_j^k$ where Q_j^k is a p*p matrix of eigenvectors and Λ_j^k is a p*p diagonal matrix of aeigenvalues. The formula can be further simplified:

$$\left[\sum_{\mathbf{i=1}}^{\mathbf{n}}\boldsymbol{\alpha_i}\emptyset(\boldsymbol{\mu_i})\right]^{\mathbf{t}}\left[\sum_{\mathbf{k=1}}^{\mathbf{K}}\sum_{\mathbf{j=1}}^{\mathbf{n_k}}J_j^k\sum_j^k(J_j^k)^t\right]\left[\sum_{\mathbf{i=1}}^{\mathbf{n}}\boldsymbol{\alpha_i}\emptyset(\boldsymbol{\mu_i})\right]$$

$$=\left[\sum_{\mathbf{i=1}}^{\mathbf{n}}\boldsymbol{\alpha_i}\emptyset(\boldsymbol{\mu_i})\right]^{\mathbf{t}}\left[\sum_{\mathbf{k=1}}^{\mathbf{K}}\sum_{\mathbf{j=1}}^{\mathbf{n_k}}J_j^k\left[Q_j^k(\Lambda_j^k)^{\frac{1}{2}}(\Lambda_j^k)^{\frac{1}{2}}Q_j^k\right](J_j^k)^t\right]\left[\sum_{\mathbf{i=1}}^{\mathbf{n}}\boldsymbol{\alpha_i}\emptyset(\boldsymbol{\mu_i})\right]$$

$$=\left[\sum_{\mathbf{i=1}}^{\mathbf{n}}\boldsymbol{\alpha_i}\emptyset(\boldsymbol{\mu_i})\right]^{\mathbf{t}}\left[\sum_{\mathbf{k=1}}^{\mathbf{K}}\sum_{\mathbf{j=1}}^{\mathbf{n_k}}J_j^k\left[\left(A_j^k\right)\left(A_j^k\right)^t\right](J_j^k)^t\right]\left[\sum_{\mathbf{i=1}}^{\mathbf{n}}\boldsymbol{\alpha_i}\emptyset(\boldsymbol{\mu_i})\right]$$

$$=\boldsymbol{\alpha^t}\left[\sum_{k=1}^{K}U_{d2k}\,U_{d2k}^t\right]\boldsymbol{\alpha}=\boldsymbol{\alpha^t}N_{d2}\boldsymbol{\alpha}$$

where $U_{d2k}=\begin{bmatrix}\sum_{\mathbf{j=1}}^{\mathbf{n_k}}\mathbf{D_{1j}^k}.A_j^k\\ .\\ .\\ .\\ \sum_{\mathbf{j=1}}^{\mathbf{n_k}}\mathbf{D_{nj}^k}.A_j^k\end{bmatrix}$ and $\mathbf{D_{ij}^k}=\emptyset(\boldsymbol{\mu_i})J_j^k=\emptyset(\boldsymbol{\mu_i}).\frac{\partial\emptyset(\mathbf{X})}{\partial\boldsymbol{\mu}_j^k}=\frac{\partial\emptyset(\boldsymbol{\mu_i}).\emptyset(\mathbf{X})}{\partial\boldsymbol{\mu}_j^k}=\frac{\partial\mathbf{K}(\boldsymbol{\mu_i},\mathbf{X})}{\partial\boldsymbol{\mu}_j^k}$

References

Aggarwal, C.C., Li, Y., Wang, J., Wang, J., 2009. Frequent pattern mining with uncertain data, in: Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. ACM, pp. 29–38.

Aggarwal, C.C., Philip, S.Y., 2009. A survey of uncertain data algorithms and applications. IEEE Transactions on Knowledge and Data Engineering 21, 609–623.

Anzanello, M.J., Albin, S.L., Chaovalitwongse, W.A., 2012. Multicriteria variable selection for classification of production batches. European Journal of Operational Research 218, 97–105.

Anzanello, M.J., Albin, S.L., Chaovalitwongse, W.A., 2009. Selecting the best variables for classifying production batches into two quality levels. Chemometrics and Intelligent Laboratory Systems 97, 111–117.

Aronszajn, N., 1950. Theory of reproducing kernels. Transactions of the American Mathematical Society 68, 337–404.

Azzalini, A., Capitanio, A., 1999. Statistical applications of the multivariate skew normal distribution. Journal of the Royal Statistical Society: Series B (Statistical Methodology) 61, 579–602.

Azzalini, A., Valle, A.D., 1996. The multivariate skew-normal distribution. Biometrika 83, 715–726.

Basseville, M., 1989. Distance measures for signal processing and pattern recognition. Signal Processing 18, 349–369.

Bhattacharyya, A., 1946. On a measure of divergence between two multinomial populations. Sankhyā: the Indian Journal of Statistics 401–406.

Bi, J., Zhang, T., 2005. Support vector classification with input data uncertainty, in: Advances in Neural Information Processing Systems. pp. 161–168.

Caliński, T., Harabasz, J., 1974. A dendrite method for cluster analysis. Communications in Statistics-theory and Methods 3, 1–27.

Cha, S.-H., 2007. Comprehensive survey on distance/similarity measures between probability density functions. City 1, 1.

Chau, M., Cheng, R., Kao, B., Ng, J., 2006. Uncertain data mining: An example in clustering location data, in: Pacific-Asia Conference on Knowledge Discovery and Data Mining. Springer, pp. 199–204.

Chernoff, H., 1952. A measure of asymptotic efficiency for tests of a hypothesis based on the sum of observations. The Annals of Mathematical Statistics 493–507.

Chiang, M.-C., Tsai, C.-W., Yang, C.-S., 2011. A time-efficient pattern reduction algorithm for k-means clustering. Information Sciences 181, 716–731.

Chuang, M.-C., Hwang, J.-N., Ye, J.-H., Huang, S.-C., Williams, K., 2017. Underwater fish tracking for moving cameras based on deformable multiple kernels. IEEE Transactions on Systems, Man, and Cybernetics: Systems 47, 2467–2477.

Cover, T.M., Thomas, J.A., 2012. Elements of information theory. John Wiley & Sons.

Csiszár, I., 1967. Information measures of difference of probability distributions and indirect observations. Studia Sci. Math. Hungar. 2, 299–318.

Csiszar, I., Körner, J., 2011. Information theory: coding theorems for discrete memoryless systems. Cambridge University Press.

Davies, D.L., Bouldin, D.W., 1979. A cluster separation measure. IEEE Transactions on Pattern Analysis and Machine Intelligence 224–227.

Devijver, P.A., Kittler, J., 1982. Pattern recognition: A statistical approach. Prentice hall.

Dorling, K., Heinrichs, J., Messier, G.G., Magierowski, S., 2017. Vehicle routing problems for drone delivery. IEEE Transactions on Systems, Man, and Cybernetics: Systems 47, 70–85.

Duda, R.O., Hart, P.E., Stork, D.G., others, 1973. Pattern classification. Wiley New York.

Dunn, J.C., 1973. A fuzzy relative of the ISODATA process and its use in detecting compact well-separated clusters.

Fraley, C., Raftery, A.E., 1998. How many clusters? Which clustering method? Answers via model-based cluster analysis. The Computer Journal 41, 578–588.

Ge, J., Xia, Y., Nadungodage, C., 2010. UNN: a neural network for uncertain data classification, in: Pacific-Asia Conference on Knowledge Discovery and Data Mining. Springer, pp. 449–460.

Gullo, F., Ponti, G., Tagarelli, A., 2010. Minimizing the variance of cluster mixture models for clustering uncertain objects, in: 2010 IEEE 10th International Conference on Data Mining (ICDM), pp. 839–844.

Gullo, F., Ponti, G., Tagarelli, A., 2008a. Clustering uncertain data via k-medoids. Scalable Uncertainty Management 229–242.

Gullo, F., Ponti, G., Tagarelli, A., Greco, S., 2017. An information-theoretic approach to hierarchical clustering of uncertain data. Information Sciences 402, 199–215.

Gullo, F., Ponti, G., Tagarelli, A., Greco, S., 2008b. A hierarchical algorithm for clustering uncertain data via an information-theoretic approach, in: 2008 IEEE 8th International Conference on Data Mining (ICDM), pp. 821–826.

Halkidi, M., Batistakis, Y., Vazirgiannis, M., 2001. On clustering validation techniques. Journal of Intelligent Information Systems 17, 107–145.

Hartigan, J.A., Wong, M.A., 1979. Algorithm AS 136: A k-means clustering algorithm. Journal of the Royal Statistical Society. Series C (Applied Statistics) 28, 100–108.

Jiang, B., Pei, J., Tao, Y., Lin, X., 2013. Clustering uncertain data based on probability distribution similarity. IEEE Transactions on Knowledge and Data Engineering 25, 751–763.

Jiao, L., Denoeux, T., Pan, Q., 2016. A hybrid belief rule-based classification system based on uncertain training data and expert knowledge. IEEE Transactions on Systems, Man, and Cybernetics: Systems 46, 1711–1723.

Kao, B., Lee, S.D., Lee, F.K., Cheung, D.W., Ho, W.-S., 2010. Clustering uncertain data using voronoi diagrams and r-tree index. IEEE Transactions on Knowledge and Data Engineering 22, 1219–1233.

Kim, S.-J., Magnani, A., Boyd, S., 2006. Optimal kernel selection in kernel fisher discriminant analysis, in: Proceedings of the 23rd International Conference on Machine Learning. ACM, pp. 465–472.

Kriegel, H.-P., Pfeifle, M., 2005. Density-based clustering of uncertain data, in: Proceedings of the Eleventh ACM SIGKDD International Conference on Knowledge Discovery in Data Mining. ACM, pp. 672–677.

Lee, S.D., Kao, B., Cheng, R., 2007. Reducing UK-means to K-means, in: Data Mining Workshops, 2007 IEEE 7th International Conference on Data Mining, pp. 483–488.

Lee, K., Kim, N., Jeong, M.K., 2014. The sparse signomial classification and regression model. Annals of Operations Research 216, 257-286.

Lichman, M., 2013. Uci machine learning repository. university of california, irvine, school of information and computer sciences.

Lissack, T., Fu, K.-S., 1976. Error estimation in pattern recognition via L_alpha-distance between posterior density functions. IEEE Transactions on Information Theory 22, 34–45.

Liu, B., Xiao, Y., Cao, L., Hao, Z., Deng, F., 2013. SVDD-based outlier detection on uncertain data. Knowledge and Information Systems 34, 597–618.

Liu, Y.-H., 2012. Mining frequent patterns from univariate uncertain data. Data & Knowledge Engineering 71, 47–68.

Liu, Z., Pan, Q., Dezert, J., Mercier, G., 2014. Credal classification rule for uncertain data based on belief functions. Pattern Recognition 47, 2532–2541.

Matusita, K., 1955. Decision rules, based on the distance, for problems of fit, two samples, and estimation. The Annals of Mathematical Statistics 631–640.

McLachlan, G., 2004. Discriminant analysis and statistical pattern recognition. John Wiley & Sons.

Mika, S., Ratsch, G., Weston, J., Scholkopf, B., Mullers, K.-R., 1999. Fisher discriminant analysis with kernels, in: Neural Networks for Signal Processing IX, 1999. Proceedings of the 1999 IEEE Signal Processing Society Workshop, pp. 41–48.

Nydick, S.W., 2012. The wishart and inverse wishart distributions. Electron. J. Statist. 6, 1–19.

O'Hagan, A., Kendall, M.G., Forster, J., 2004. Kendall's Advanced Theory of Statistics: Bayesian Statistics. Vol. 2B. Arnold.

Pakhira, M.K., Bandyopadhyay, S., Maulik, U., 2005. A study of some fuzzy cluster validity indices, genetic clustering and application to pixel classification. Fuzzy Sets and Systems 155, 191–214.

Pakhira, M.K., Bandyopadhyay, S., Maulik, U., 2004. Validity index for crisp and fuzzy clusters. Pattern Recognition 37, 487–501.

Pal, N.R., Biswas, J., 1997. Cluster validation using graph theoretic concepts. Pattern Recognition 30, 847–857.

Peel, M.C., Finlayson, B.L., McMahon, T.A., 2007. Updated world map of the Köppen-Geiger climate classification. Hydrology and Earth System Sciences Discussions 4, 439–473.

Qin, B., Xia, Y., Li, F., 2010. A Bayesian classifier for uncertain data, in: Proceedings of the 2010 ACM Symposium on Applied Computing. ACM, pp. 1010–1014.

Qin, B., Xia, Y., Li, F., 2009a. DTU: a decision tree for uncertain data, in: Pacific-Asia Conference on Knowledge Discovery and Data Mining. Springer, pp. 4–15.

Qin, B., Xia, Y., Prabhakar, S., Tu, Y., 2009b. A rule-based classification algorithm for uncertain data, in: 2009 IEEE 25th International Conference on Data Engineering (ICDE), pp. 1633–1640.

Qin, X., Zhang, Y., Li, X., Wang, Y., 2010. Associative classifier for uncertain data, in: International Conference on Web-Age Information Management. Springer, pp. 692–703.

Rauber, T.W., Braun, T., Berns, K., 2008. Probabilistic distance measures of the Dirichlet and Beta distributions. Pattern Recognition 41, 637–645.

Ren, J., Lee, S.D., Chen, X., Kao, B., Cheng, R., Cheung, D., 2009. Naive bayes classification of uncertain data, in: 2009 IEEE 9th International Conference on Data Mining (ICDM), pp. 944–949.

Rousseeuw, P.J., 1987. Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. Journal of Computational and Applied Mathematics 20, 53–65.

Shin, K.S., Jeong, Y.S., Jeong, M.K., 2012. A two-leveled symbiotic evolutionary algorithm for clustering problems. Applied Intelligence 36, 788-799.

Tavakkol, B., Jeong, M.K., Albin, S.L., 2017. Object-to-group probabilistic distance measure for uncertain data classification. Neurocomputing 230, 143–151.

Tsang, S., Kao, B., Yip, K.Y., Ho, W.-S., Lee, S.D., 2011. Decision trees for uncertain data. IEEE Transactions on Knowledge and Data Engineering 23, 64–78.

Xie, X.L., Beni, G., 1991. A validity measure for fuzzy clustering. IEEE Transactions on Pattern Analysis and Machine Intelligence 13, 841–847.

Xu, D., Albin, S.L., 2006. Optimizing settings by accounting for uncontrollable material and environmental variables. IIE Transactions 38, 1085–1092.

Xu, L., Hung, E., 2011. Distance-based feature selection on classification of uncertain objects, in: Australasian Joint Conference on Artificial Intelligence. Springer, pp. 172–181.

Yang, B., Zhang, Y., 2010. Kernel based K-medoids for clustering data with uncertainty. Advanced Data Mining and Applications 246–253.

Yang, J., Gunn, S., 2007. Exploiting uncertain data in support vector classification, in: International Conference on Knowledge-Based and Intelligent Information and Engineering Systems. Springer, pp. 148–155.

Young, S.H., Mazzuchi, T.A., Sarkani, S., 2017. A Framework for Predicting Future System Performance in Autonomous Unmanned Ground Vehicles. IEEE Transactions on Systems, Man, and Cybernetics: Systems 47, 1192–1206.

Zhang, H., Albin, S., 2007. Determining the number of operational modes in baseline multivariate SPC data. IIE Transactions 39, 1103–1110.

Zhang, H., Albin, S.L., Wagner, S.R., Nolet, D.A., Gupta, S., 2010. Determining statistical process control baseline periods in long historical data streams. Journal of Quality Technology 42, 21–35.

Zhou, S., Chellappa, R., 2004. Probabilistic distance measures in reproducing kernel Hilbert space. SCR Technical Report, University of Maryland.