

Improving Classification Accuracy on Uncertain Data by Considering Multiple Subclasses

Lei Xu and Edward Hung

Department of Computing, The Hong Kong Polytechnic University
{cslxu, csehung}@comp.polyu.edu.hk

Abstract. We study the problem of classification on uncertain objects whose locations are uncertain and described by probability density functions (pdf). Though there exist some classification algorithms proposed to handle uncertain objects, all existing algorithms are complex and time consuming. Thus, a novel supervised UK-means algorithm is proposed to classify uncertain objects more efficiently. Supervised UK-means assumes the classes are well separated. However, in real data, subsets of objects of the same class are usually interspersed among (disconnected by) other classes. Thus, we proposed a new algorithm Supervised UK-means with Multiple Subclasses (SUMS) which considers the objects in the same class can be further divided into several groups (subclasses) within the class and tries to learn the subclass representatives to classify objects more accurately.

Keywords: classification, supervised UK-means, multiple subclasses, uncertain objects.

1 Introduction

While there has been a large amount of research on mining and queries on relational databases [15], the focus has been on databases that store data in exact values. In many real-life applications, however, the raw data (for example, in the case of sensor data) are not precise or accurate when they were collected or produced. Data uncertainty are classified into two types. One is existential uncertainty caused by not sure the existence of an object or a data tuple [1, 4, 5, 14]; The other is value uncertainty caused by not knowing the value precisely. In this paper, we focus on the second case (value uncertainty).

Classification is an important task in machine learning. [18, 19] shows that the learning based similarity metrics can perform better in multimedia data classification and retrieval. Similarly to [18, 19], we learn model based expected Euclidean distance used in UK-means. In previous work, we modified UK-means to supervised UK-means to classify uncertain objects in [21]. We considers the problem of classifying objects as a multi-dimensional uncertainty where an object is represented by an uncertain region over which a discrete probability distribution function (PDF), or a probability density function (pdf), is defined. Formally, we consider a training set of N labeled objects o_i , $1 \leq i \leq N$ in an m -dimensional

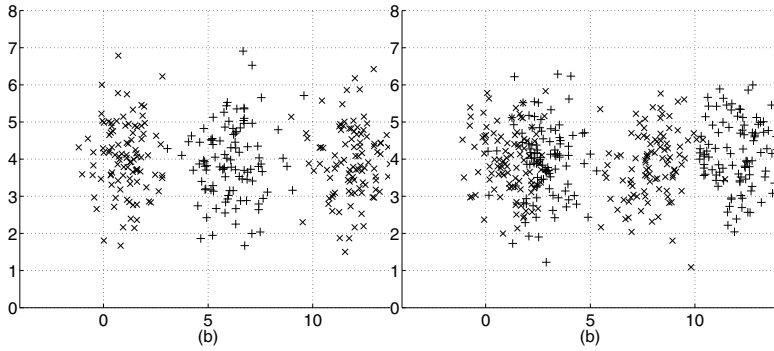


Fig. 1. (a) An example of one class ('x') divided by another class ('+') (b) Another example of one class ('+') divided by another class ('x')

space. An object o_i is represented by a pdf $f_i: IR^m \rightarrow IR$ (IR represents real number space) that specifies the probability density of each possible location of object o_i ¹. Thus, each object can be bounded by a finite bounding box. This assumption is reasonable because in practice the probability density of an object is high only within a very small region of concentration. o_i is associated with a class label c_j ($c_j \in L$), where L is a class label set.

Supervised UK-means assumes the classes are well separated. However, in classification, the classes are often in arbitrary shape which makes the boundary between classes concave or convex but not a single line. In some cases, a class's objects are separated (disconnected) by objects of other classes. As Figure 1(a) shows one class (represented by 'x') is divided by another class ('+'). Figure 1(b) is another example showing that each class is divided by other classes and the boundary between classes is concave and convex. For the above cases, our solution is using a few class representatives to represent a class. We consider the problem as the estimation of the number of groups or subclasses (k) in each class. The key idea of estimation of k is to use splitting and/or merging methods to increase and/or decrease the number of clusters, which makes the model fit the data. Each subclass (cluster belonging to a class) can be considered as a Gaussian mixture model. Several algorithms have been proposed to determine k automatically [7, 8, 11, 20]. Most of them wrap around either K-means or Expectation-Maximization for fixed k . In a classification problem, training objects are labeled by class labels. When we consider objects from a class and further divide them into subclasses, we ignore their labels. In other words, objects are labeled during inter-class training, and in the process of the estimation of k (the number of subclasses in a class) during intra-class training, the labels of objects from the same class are ignored.

¹ The methods to be discussed in this paper require that for each object o_i , the uncertain region $UD(o_i)$ of each object o_i is finite, i.e. $\forall x \notin UD(o_i), f_i(x) = 0$.

2 Related Work

A large number of classification models have been proposed in the literature. However, few research focuses on the problem of classification handling uncertain data caused by value uncertainty. In [2], support vector machine is used to classify uncertain data. In this method, an uncertain object is assumed as a simple bounded geometric model. Support vector machine creates margins by using uncertain objects which lie on the boundary. In this model, the uncertainty (domain of uncertain objects) of objects are estimated. In our model, the maximum boundary of objects is fixed.

In [16, 17], an uncertain object is associated with a probability density function (pdf) and a finite region. The decision tree classifier is extended to handle uncertain data by using averaging or distribution-based approach. To improve the efficiency, some pruning techniques were proposed without affecting the results of decision tree. In [13], the uncertain model is the same as that in [16, 17]. In [13], Naive Bayes is used to classify uncertain data. Three approaches are proposed (averaging, formula-based, and sample-based) to calculate the probability of object label and assign it to the class with highest probability. In [12], uRule is proposed based on Rule-based algorithm to classify uncertain information. The difference between Rule-based and uRule is that the instances are partly covered by the rule in uRule. The key idea in uRule is that the algorithm computes which proportion of the instances is covered by a rule based on the uncertain attribute interval and probabilistic function. uRule considers to classify uncertain numerical and categorical data. Though some algorithms have been extended to classify uncertain information, the problem of building classifiers on uncertain data is still a challenge. The algorithms take quite a long time to classify uncertain objects because they are too complex.

Some work has been proposed to estimate the number of clusters during data clustering. X-means [11] is a regularization framework for learning k with K-means. X-means tries many values of k and obtains a model for each k value. Then X-means uses the Bayesian Information Criterion (BIC) to score each model and chooses the model with the highest BIC score. Other scoring criteria are also used in X-means. The drawback of X-means is that the algorithm assumes the cluster covariances are all spherical with the same width. X-means is likely to overfit the data when the clusters are non-spherical. Bayesian K-means [20] uses Maximization-Expectation (ME) to learn a mixture model. ME maximizes over the hidden variables (assignment of examples to clusters), and computes an expectation over model parameters (center location and covariances). The algorithm works well but it is time consuming. G-means (Gaussian means) [8] is a wrapper around the K-means algorithm. G-means uses projection and a statistical test for the hypothesis that the data in a cluster come from a Gaussian distribution. The algorithm grows k with a small number of centers. It applies a statistical test to each cluster and those which are not accepted as Gaussian are split into two clusters. Interleaved with K-means, this procedure repeats until every cluster's data are accepted as Gaussian. The method does not assume spherical clusters, but it works well when true clusters are well-separated.

It has difficulty when true clusters overlap since the hard assignment of K-means can clip data into subsets that look non-Gaussian. PG-means (Projected Gaussian means) [7] is proposed to handle difficult cases, such as non-Gaussian data, overlapping clusters, eccentric clusters, high dimension. Moreover, PG-means is faster than variational Bayesian K-means. PG-means is based on projections and statistical tests to determine whether a whole mixture model fits the data well. PG-means cannot be directly used in our UK-means classification, because PG-means is used to handle certain objects. In Section 3.4, we will describe how to modify PG-means for our problem.

3 Supervised UK-Means

3.1 Problem Definition

In the supervised model, there are a set of N training objects $o_1, o_2, \dots, o_N$, and m numerical (real-valued) feature attributes $A_1, \dots, A_m$. The domain of attribute A_u ($1 \leq u \leq m$) is $\text{dom}(A_u)$. Each o_i is associated with a probability density function (pdf $f_i(x)$) and a class label c_j ($c_j \in L$, where L is the set of all class labels), where x is a possible location of o_i , and $UD(o_i)$ is uncertain domain of o_i . Each tuple x is associated with a feature vector $x = (\tilde{x}_1, \tilde{x}_2, \dots, \tilde{x}_m)$, where $\tilde{x}_u \in \text{dom}(A_u)$ ($1 \leq u \leq m$). The goal of supervised UK-means is to find K (the number of labels, $|L|$) class representatives which can predict a testing object o_{test} to class c_k with the minimum expected Euclidean distance. The boundary between classes are sometimes in arbitrary shape (e.g. convex or concave). Also, the objects from the same class may be disconnected by other classes. It is obvious that one class representative is not enough for representing a class. In our model, we assume one class can be further divided into a number subclasses, and we try to estimate an appropriate number of subclasses of each class. Then, we find the subclasses' representatives to predict the labels of testing objects with the minimum expected Euclidean distance between objects and subclass representatives.

3.2 Expected Euclidean Distance

As Figure 2 shows, one way to approximate the calculation is to divide the uncertain domain into a number of grid cells. A sample is considered to be located at the center of each grid cell which represents a possible location of the uncertain object o_i . The expected Euclidean distance (EED) from object o_i (represented by a pdf f_i) to the cluster representative p_{c_j} is the weighted average of the distances between the samples in o_i and $\bar{p}_{c_j}$ (the mean vector of p_{c_j}), i.e. $EED(o_i, p_{c_j}) = \sum_{t=1}^T F_i(s_{i,t}) ED(s_{i,t}, \bar{p}_{c_j})$, where T is the number of samples in o_i , $s_{i,t}$ is the location (vector) of the t -th sample of o_i , p_{c_j} is a probabilistic object representing the domain of the cluster representative of cluster c_j , $F_i(s_{i,t}) = \int_{x \in \text{cell}_t} f_i(x) dx$ (F_i is a discrete probability distribution function over T grid cells, cell_t is the grid cell that sample $s_{i,t}$ represents, x is

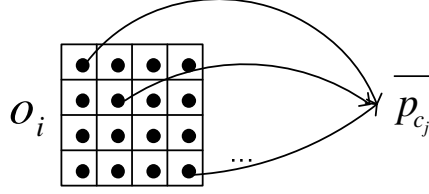


Fig. 2. Expected distance calculation from o_i to p_{c_j} in [3, 9, 10]

the possible location of sample $s_{i,t}$ in $cell_t$), and the metric ED is Euclidean distance used in [3, 9, 10]. In this paper, the mean vector $\overline{o_i}$ (expected value) of o_i , which is the weighted mean of all T samples (or possible locations), is calculated as Equation (1).

$$\overline{o_i} = \sum_{t=1}^T s_{i,t} \times F_i(s_{i,t}). \quad (1)$$

The mean vector $\overline{p_{c_{j_k}}}$ of k -th subclass representative $p_{c_{j_k}}$ of class c_j is obtained by Equation (2), where $|c_{j_k}|$ is the number of objects assigned to subclass c_{j_k} , and $C(o_i) = c_{j_k}$ means that object o_i is assigned to subclass c_{j_k} .

$$\overline{p_{c_{j_k}}} = \frac{1}{|c_{j_k}|} \sum_{o_i \in \{o_i | C(o_i) = c_{j_k}\}} \overline{o_i}. \quad (2)$$

3.3 Supervised UK-Means with Multiple Subclasses (SUMS)

Algorithm 1 shows Supervised UK-means with Multiple Subclasses (SUMS), where N is the number of training objects, K is the number of class labels, k_l is the number of subclasses of the l -th class. In Algorithm 1, the number of subclass representatives is estimated and the subclass representatives are trained by the objects of the same class. First, Algorithm 1 calculates the mean vectors of uncertain objects. Then the number of subclasses of each class is estimated by Algorithm 2. The subclass representatives are trained by UK-means based on the objects of the same class. The time complexity of the algorithm is decided by the time costing on estimation of k_i (the number of subclasses) and the calculation of subclass representatives. The time complexity of training subclass representatives is $O(NTKk_{max})$, where N is the number of training objects, T is the number of samples of object, K is the number of class labels, and k_{max} is the maximum number of subclasses of a class. The time complexity of estimation of k_i (the number of subclasses) will be discussed in Section 3.4.

3.4 Estimation of k_i

In [7], PG-means uses a statistical hypothesis test on one-dimensional projection of the data and model to determine if the examples are well represented by the

Algorithm 1. Supervised UK-means with Multiple Subclasses (SUMS)Input: training set $\{o_1, o_2, \dots, o_N\}$ with class labels c_j ($j \in \{1, 2, \dots, K\}$)

Output: learner of sub-class representatives.

```

1: for  $i = 0; i < N; i++$  do
2:   compute  $\bar{o}_i$  of training objects by Equation (1);
3: end for
4: for  $i=0; i < K; i++$  do
5:   Estimate the number of subclasses  $k_i$  by PG-means and get the subclass rep-
     resentatives  $(p_{c_{i_1}}, p_{c_{i_2}}, \dots, p_{c_{i_{k_i}}})$ ;
6: end for
7: repeat
8:   for  $m=0; m < N; m++$  do
9:     for  $i=0; i < K; i++$  do
10:      for  $j=0; j < k_i; j++$  do
11:        compute Expected Euclidean Distance by  $EED(o_m, p_{c_{i_j}}) =$ 
           $\sum_{t=1}^T F_m(s_{m,t})ED(s_{m,t}, p_{c_{i_j}})$ ;
12:      end for
13:    end for
14:    assign object  $o_m$  to the subclass with the minimum  $EED(o_m, p_{c_{i_q}})$ ;
15:  end for
16:  update all subclass representatives by Equation (2);
17: until all subclass representatives converge

```

model. PG-means stands for Projected Gaussian and refers to the fact that the method applies the projections to the clustering model as well as the data, before performing each hypothesis test for model fitness. In [7], the standard Gaussian mixture model is used with Expectation-Maximization training. The key idea of PG-means is using the EM algorithm to learn a model containing k centers. When EM learning converges, PG-means projects both the data set and the learned model to one dimension, and then applies the Kolmogorov-Smirnov (KS) test to determine whether the projected model fits the projected data.

Assume some data X is sampled from a single Gaussian cluster with distribution $X \sim (\mu, \Sigma)$ in d dimension. $\mu = E[X]$ is the $d \times 1$ mean vector and $\Sigma = cov[X]$ is the $d \times d$ covariance matrix. Given a $d \times 1$ projection vector P of the unit length ($\|P\| = 1$), we can project X along P as $X' = P^T X$. Then, $X' \sim N(\mu', \sigma)$, where $\mu' = P^T \mu$ and $\sigma^2 = P^T \Sigma P$. We can project each cluster model to obtain a one-dimensional projection of an entire mixture along P . PG-means [7] repeats this projection and test step several times for a single learned model. If any test rejects the null hypothesis that the data follows the model's distribution, then it adds one subclass and starts with UK-means in our algorithm. If every test accepts the null hypothesis for a given model, then the algorithm terminates. The following are the two hypotheses:

H_0 : The data around the center are sampled from a Gaussian.

H_1 : The data around the center are not sampled from a Gaussian.

Algorithm 2. Estimation of k_i

Input: The mean vectors of objects X with the same label c_i , confidence α , number of projections J .

Output: the number of subclasses (k_i) of class c_i and subclass representatives $(p_{c_{i_1}}, p_{c_{i_2}}, \dots, p_{c_{k_i}})$.

- 1: Let $k_i \leftarrow 1$. Initialize the class with the mean and covariance of X .
 - 2: **for** $j = 0; j < J; j++$ **do**
 - 3: Randomly generate a $d \times 1$ projection vector P .
 - 4: Project X and the model to P with the same projection.
 - 5: Use KS test at significance level α to test if the projected model fits the projected data set.
 - 6: **if** any test rejects the null hypothesis **then**
 - 7: break out of the loop.
 - 8: **end if**
 - 9: **end for**
 - 10: **if** any test rejected the null hypothesis **then**
 - 11: Initialize $k_i + 1$ subclasses as the k_i previously learned plus one new subclass.
 - 12: Run UK-means on $k_i + 1$ subclasses to learn $k_i + 1$ subclass representatives.
 - 13: Let $k_i \leftarrow k_i + 1$, and go to step 2.
 - 14: **end if**
 - 15: Every test accepts the null hypothesis; stop and return the model.
-

After projection, PG-means uses the univariate Kolmogorov-Smirnov (KS) test for model fitness. The KS test statistic is $D = \max_X |F(X) - S(X)|$ —the maximum absolute difference between the true cumulative distribution function (CDF) of $F(x)$ with the sample CDF $S(X)$. Algorithm 2 shows the algorithm more formally. PG-means uses UK-means to learn a model containing k_i centers. In our work, we use UK-means instead of EM training [7]. UK-means uses hard assignment (each example has membership in only one subclass). The worst case is that each object belongs to a subclass, and the time complexity of Algorithm 2 is $O(Jn^2T)$, where J is the number of projections, n is the number of objects of a class, T is the number of samples. In the algorithm, we use random projection [6] to project the data and the model. There are also other possible methods, e.g. principal component analysis which can be used to do projection. We wish to use a small but sufficient number of projections and tests to discover when a model does not fit data well. In [7], they followed Dasgupta’s conclusion that c -separation² is the natural measure for Gaussian [6]. In [6, 7], they gave the conclusion that if J random projections are performed, the probability that all J projections are ‘bad’³ is less than some ϵ : $Pr(J \text{ bad projections}) = \text{Erf}(\sqrt{1/2})^J < \epsilon$, where Erf is the standard Gaussian error function. To find

² For any $c > 0$, assume the two cluster centers μ_1 and μ_2 are in d dimension and the spherical covariances Σ of the two clusters are the same for simplicity, c -separation is that the vector m connecting the two centers ($m = \mu_2 - \mu_1$) satisfies the condition $\|m\| \geq c\sqrt{\text{trace}(\Sigma)}$.

³ The probability that J is a ‘bad’ projection, i.e. that it does not maintain c -separation between the cluster means when projected.

a projection that keeps the two clusters means c -separated, approximately J is shown as $J < \log(\epsilon)/\log(\text{Erf}(\sqrt{1/2})) \approx -2.6198\log(\epsilon)$ follows [6, 7]. For example, if ϵ is 0.01, 12 projections are needed. In the experiments, we use $J = 12$ projections to estimate the number of subclasses in a class.

4 Experimental Evaluation

In this section, we evaluate Supervised UK-means with Multiple Subclasses (SUMS) by comparing it with supervised UK-means. We compare their classification results in this section. All algorithms were written in Matlab and were run on a Windows machine with an Intel 2.66GHz Pentium(R) Dual-Core processor and 4GB of main memory.

4.1 Synthetic Data Sets

We have done the experiments on the two typical data sets shown in Figure 1(a) and Figure 1(b). In Figure 1(a), the class ('x') is divided by the class ('+'). The number of objects in class ('+') is 100, and the number of objects in class ('x') is 200 with equally distribution on the two sides of class ('+'). We denote the data set by "Middle" which means that one class is divided in the middle. The centers of uncertain objects were generated from a Gaussian distribution, whose mean and standard deviation are equal to the class center and σ respectively. A set of uncertain objects represented by MBRs with size $S \times S$ were randomly generated in $2D$ space $[-100, 100] \times [-100, 100]$. An MBR is divided into $\sqrt{T} \times \sqrt{T}$ grid cells. Each grid cell corresponds to a sample. Each sample is associated with a randomly generated probability value. All probabilities in an MBR are normalized to have their sum equal to 1. For each data set, a set of K class representatives were trained by the labeled objects. Similarly, in Figure 1(b), the number of objects in class ('x') is 200, and the number of objects in class ('+') is 200. The class 'x' and the class '+' are divided into two subclasses by each other. We denote the example by "Side" with the meaning that each class being divided on two sides. Each subclass has 100 objects. The execution time is shown in Table 2. SUMS spends more time than supervised UK-means. Because in SUMS, PG-means is used to find the subclasses of class which costs extra time. The time of SUMS depends on the number of training

Table 1. Parameters for experiments using data sets

Parameter	Description	Baseline Value
K	number of classes	2
T	number of samples per object	49
w	maximum size of MBR, $S \times S$	0.25
D	number of dimensions	2
σ	standard deviation of Gaussian distribution	1

Table 2. Execution Time (Seconds)

Data Set	Supervised UK-means	SUMS
Middle	0.589	1.988
Side	0.704	3.215

objects, the number of samples and the number of projections. Table 3 shows that the accuracy is improved by SUMS compared with supervised UK-means. In Figures 3 and 4, the (symbol ‘o’ is the learned (sub)class representative of a class). In Figure 1(a), the center of class ‘x’ is (1, 4) and (12, 4), and the center of class ‘+’ is (6, 4). As Figure 3(a) shows, in supervised UK-means, the learned mean vector of class ‘x’ is (2.7931, 4.2881) while the learned mean vector of class ‘+’ is (10.9975, 4.2332). Figure 3(b) shows, in SUMS, the learned class representatives of class ‘+’ is (6.6879, 4.2589) while the learned class representatives of class ‘x’ is (1.1130, 4.2304) and (12.0707, 4.2947). In Figure 1(b), the subclass representatives of class ‘x’ is (1, 4) and (8, 4) and the center of class ‘+’ is (3, 4) and (12, 4). The learned subclass representatives of class ‘x’ are (0.8984, 4.0726) and (8.4050, 4.0098) while the learned subclass representatives of class ‘+’ are (3.1032, 3.9298) and (12.1603, 3.8576) (Figure 4(b)) in SUMS. In Figure 4(a), the supervised UK-means can only learn one class representative for each class (the class representative of class ‘x’ is (1.8759, 4.0096) and the class representative of class ‘+’ is (10.2837, 3.9341)). In Figure 3(a) and 4(a), the learned class representative is far from some objects from the same class which makes the accuracy of classifier low. In Figure 3(b) and 4(b), the learned

Table 3. Accuracy on synthetic data sets

Data Set	Supervised UK-means	SUMS
Middle	0.4667	0.9833
Side	0.5175	0.9067

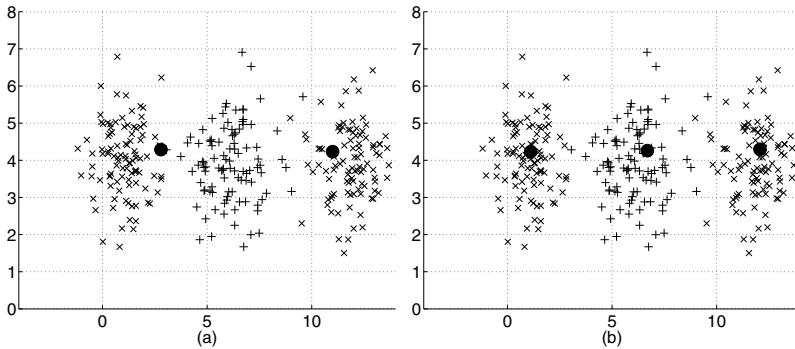


Fig. 3. (a) Class representatives of “Middle” learned by Supervised UK-means (b) Subclass representatives of “Middle” learned by SUMS

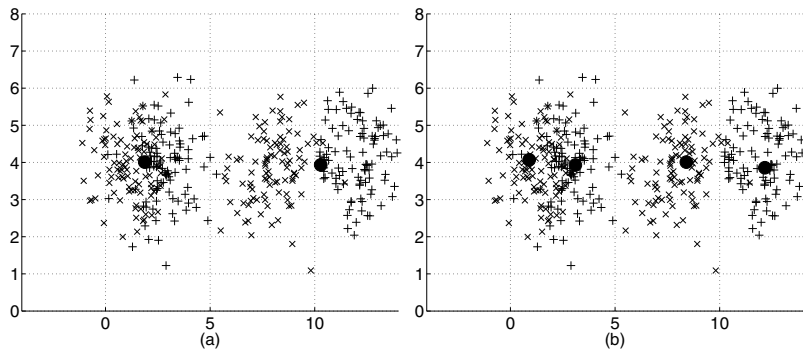


Fig. 4. (a) Class representatives of “Side” learned by Supervised UK-means (b) Subclass representatives of “Side” learned by SUMS

subclass representatives can improve the performance of supervised UK-means, because the subclass representatives are closer to their real class centers. From the experiments, the SUMS algorithm can learn more information compared with supervised UK-means. SUMS with PG-means can learn the subclasses of the synthetic data sets. SUMS improves the accuracy of supervised UK-means with more time cost. We also did experiments on other synthetic data sets which also show that SUMS can learn subclasses more accurately than supervised UK-means. From the experiments, if the number of subclasses (as well as classes, samples, objects, or dimensions) increases, the time of SUMS will increase. Because of space limitation, we will not show the figures.

4.2 Real Data Sets

We have also done experiments on real data sets. The parameters of the chosen data sets used for the experiments are summarized in Table 4. The attributes of all the data sets are numerical obtained from measurements. Classifiers are built on the numerical attributes and their “class label” attributes. For the chosen data sets, we use 10-fold cross validation to measure the accuracy. The 3 data sets contains “point values” without uncertainty. Similar to uncertainty generation of synthetic data sets, each object is represented by an MBR with size 0.25×0.25 in a multiple dimension space, which is divided into $\sqrt{49} \times \sqrt{49}$ grid cells. Each grid cell corresponds to a sample. Each sample is associated with a randomly

Table 4. Selected Data Sets from the UCI Machine Learning Repository

Data Set	Training Tuples	No. of Attributes	No. of Classes	Test Tuples
Iris	150	4	3	10-fold
BreastCancer	569	30	2	10-fold
Ionosphere	351	32	2	10-fold

Table 5. Accuracy on real data sets

Data Set	Supervised UK-means	SUMS
Iris	0.913	0.927
BreastCancer	0.843	0.895
Ionosphere	0.706	0.871

generated probability value. All probabilities in an MBR are normalized to have their sum equal to 1. From Table 5, SUMS with PG-means can classify the objects more accurately on real data sets compared with supervised UK-means. Supervised UK-means learns one class representative for each class. However, the objects of a class may be distributed nearer to other class representatives. Thus, the SUMS can train more than one class representatives for each class by PG-means. In SUMS, PG-means tries to estimate the number of subclasses of a class and learn local subclass representatives which may be closer to the objects belonging to the same class. The experiments on both synthetic and real data sets show that the Supervised UK-means with Multiple Subclasses (SUMS) can improve the accuracy of supervised UK-means.

5 Conclusion

In our work, we study the problem of classification on uncertain objects whose locations are uncertain and described by probability density functions (pdf). Though supervised UK-means algorithm is proposed to classify uncertain objects more efficiently, it assumes that the classes are well separated. In real data, subsets of objects of the same class are usually interspersed among (disconnected by) other classes. Thus, we consider the objects in the same class can be further divided into several groups (subclasses) within the class. We use PG-means (projected Gaussian) to estimate the number of subclasses in the class and train the subclass representatives by UK-means. Our experimental results demonstrated that our algorithm (SUMS) can overcome the limitation of supervised UK-means when a class is divided by other classes.

Acknowledgement. The work described in this paper was partially supported by grants from the Research Grants Council of the Hong Kong Special Administrative Region, China (PolyU 5182/08E, PolyU 5191/09E).

References

1. Barbará, D., Garcia-Molina, H., Porter, D.: The management of probabilistic data. *IEEE Trans. Knowl. Data Eng.* 4(5), 487–502 (1992)
2. Bi, J., Zhang, T.: Support vector classification with input data uncertainty. In: *NIPS* (2004)

3. Chau, M., Cheng, R., Kao, B., Ng, J.: Uncertain Data Mining: An Example in Clustering Location Data. In: Ng, W.-K., Kitsuregawa, M., Li, J., Chang, K. (eds.) PAKDD 2006. LNCS (LNAI), vol. 3918, pp. 199–204. Springer, Heidelberg (2006)
4. Cheng, R., Kalashnikov, D.V., Prabhakar, S.: Querying imprecise data in moving object environments. *IEEE Trans. Knowl. Data Eng.* 16(9), 1112–1127 (2004)
5. Dalvi, N.N., Suciu, D.: Efficient query evaluation on probabilistic databases. *VLDB J.* 16(4), 523–544 (2007)
6. Dasgupta, S.: Experiments with random projection. In: *UAI*, pp. 143–151 (2000)
7. Feng, Y., Hamerly, G.: Pg-means: learning the number of clusters in data. In: *NIPS*, pp. 393–400 (2006)
8. Hamerly, G., Elkan, C.: Learning the k in k -means. In: *NIPS* (2003)
9. Kao, B., Lee, S.D., Cheung, D.W., Ho, W.-S., Chan, K.F.: Clustering uncertain data using voronoi diagrams. In: *ICDM*, pp. 333–342 (2008)
10. Ngai, W.K., Kao, B., Chui, C.K., Cheng, R., Chau, M., Yip, K.Y.: Efficient clustering of uncertain data. In: *ICDM 2006*, pp. 436–445 (2006)
11. Pelleg, D., Moore, A.W.: X-means: Extending k -means with efficient estimation of the number of clusters. In: *ICML*, pp. 727–734 (2000)
12. Qin, B., Xia, Y., Prabhakar, S., Tu, Y.-C.: A rule-based classification algorithm for uncertain data. In: *ICDE*, pp. 1633–1640 (2009)
13. Ren, J., Lee, S.D., Chen, X., Kao, B., Cheng, R., Cheung, D.W.-L.: Naive bayes classification of uncertain data. In: *ICDM*, pp. 944–949 (2009)
14. Ruspini, E.H.: A new approach to clustering. *Information and Control* 15(1), 22–32 (1969)
15. Tan, P.-N., Steinbach, M., Kumar, V.: *Introduction to Data Mining*. Addison-Wesley (2005)
16. Tsang, S., Kao, B., Yip, K.Y., Ho, W.-S., Lee, S.D.: Decision trees for uncertain data. In: *ICDE*, pp. 441–444 (2009)
17. Tsang, S., Kao, B., Yip, K.Y., Ho, W.-S., Lee, S.D.: Decision trees for uncertain data. *IEEE Trans. Knowl. Data Eng.* 23(1), 64–78 (2011)
18. Wang, D., Kim, Y.-S., Park, S.C., Lee, C.S., Han, Y.K.: Learning based neural similarity metrics for multimedia data mining. *Soft Comput.* 11(4), 335–340 (2007)
19. Wang, D., Ma, X.: Learning Pseudo Metric for Multimedia Data Classification and Retrieval. In: Negoita, M.G., Howlett, R.J., Jain, L.C. (eds.) *KES 2004*. LNCS (LNAI), vol. 3213, pp. 1051–1057. Springer, Heidelberg (2004)
20. Welling, M., Kurihara, K.: Bayesian k -means as a “maximization-expectation” algorithm. In: *SDM* (2006)
21. Xu, L., Hung, E.: Distance-Based Feature Selection on Classification of Uncertain Objects. In: Wang, D., Reynolds, M. (eds.) *AI 2011*. LNCS, vol. 7106, pp. 172–181. Springer, Heidelberg (2011)