A Naïve Bayesian Classifier in Categorical Uncertain Data Streams

Jiaqi Ge and Yuni Xia
Department of Computer and Information Science
Indiana University and Purdue University Indianapolis
Indianapolis, Indiana, USA, 46202
Email: {jiaqge,yxia}@cs.iupui.edu

Jian Wang
School of Electronic Science and Engineering
Nanjing University
Nanjing, Jiangsu, China, 210023
Email:wangjnju@nju.edu.cn

***Abstract*—This paper proposes a novel naïve Bayesian classifier in categorical uncertain data streams. Uncertainty in categorical data is usually represented by vector valued discrete pdf, which has to be carefully handled to guarantee the underlying performance in data mining applications. In this paper, we map the probabilistic attribute to deterministic points in the Euclidean space and design a distance based and a density based algorithms to measure the correlations between feature vectors and class labels. We also devise a new pre-binning approach to guarantee bounded computation and memory cost in uncertain data streams classification. Experimental results in real uncertain data streams prove that our density-based naive classifier is efficient, accurate, and robust to data uncertainty.**

I. Introduction

Data streams are widely used to model data in a lot of applications such as sensor networks, RFID networks, and network monitoring systems. Data uncertainty, which makes data imprecise or misleading, originates from many sources as data collection error, measurement precision limitations, sampling error, and transmission error. For example, in the traffic surveillance system, because we only record the state of a vehicle at the recording time, the exact state of the vehicle at any other time can only be inferred from data probabilistically. When the state of the vehicle is categorical like normal/abnormal, the uncertain state of the vehicle can be represented by a discrete pdf as normal:0.4, abnormal:0.6. A categorical attribute is an attribute with finite possible values, and a categorical uncertain attribute is a categorical attribute whose value is probabilistic. Data streams with categorical uncertain attributes are so called categorical uncertain data streams.

Data uncertainty has to be carefully managed; otherwise it would significantly downgrade the underlying performance of various data mining applications. A typical approach is to use the expectation of the probabilistic attribute to manage data uncertainty [1], [2]. However, the expectation is only one of the statistic observations of uncertain data, and lots of useful information is lost if we simply use the expectation to model uncertain data. Some other methods adopt the possible world semantic [3], [4] to enumerate all possible databases to analyze uncertain data. Although these methods lead to an accurate result, it is usually too complex to be used in uncertain data stream mining, because of its exponential computational complexity.

We develop a new approach to manage data uncertainty to induce naïve Bayesian classifier in categorical uncertain data streams. We believe that data in one class are similar in some aspects, which is the reason why they are classified to the same class. In order to analyze the relation between the uncertain features of a data instance and its class label, we innovatively map uncertain attribute values to data points in the Euclidean space, where the coordinates of data points are transformed from vector-valued pdfs. Classification model is a function which maps the features into class labels. In previous uncertain classifiers [2], [5]–[7], uncertain attribute values are treated as random variables, and the classification model in uncertain data maps possible values of the uncertain feature into class labels. However, this approach has the exponential complexity, and various assumptions are made to make it practical. Meanwhile, the model is obscure to be understood because of its probabilistic intrinsics. However, by mapping the pdfs into data points in the Euclidean space, it helps to reveal the relationship between the space of pdfs and the space of class labels, and also helps to understand the classification rules by a new insight of the data uncertainty.

Data stream classification is very sensitive to both memory and computation cost, because data are coming continuously with a fast rate. Typically, the algorithm can only scan the data in one-pass, and this one-pass constraint dictates the choice of data structures and algorithms that can be used in data stream classification [8]. Meanwhile, managing pdf-represented uncertain data usually requires more computational resources, comparing to that in mining certain data. These new challenges bring the uncertain data stream mining problems up to the front recently.

In this paper, we propose a novel algorithm to induce the naïve Bayesian classifier in categorical uncertain data streams. Our main contributions are listed as follows.

- We model the uncertain categorical data streams by mapping the pdfs of uncertain categorical attribute to data points in the Euclidean space, and estimate the density of points by the multi-dimensional kernel density estimator.
- We build a distance based and a density based naïve Bayesian model to classifying uncertain categorical data streams.

This work is partially supported by the Research programs of Jiangsu Province (No. BY2014126-2)

This is the author's manuscript of the article published in final edited form as:
Ge, J., Xia, Y., & Wang, J. (2014). A Naive Bayesian classifier in categorical uncertain data streams. In 2014 International Conference on Data Science and Advanced Analytics (DSAA) (pp. 392–398). http://doi.org/10.1109/DSAA.2014.7058102

TABLE I. AN EXAMPLE OF THE CATEGORICAL UNCERTAIN DATA STREAM

Id	Color	Class
1	Red:0.2, Green:0.6, Blue:0.2	1
2	Red:0.6, Green:0.2, Blue:0.2	2
3	Red:0.2, Green:0.5, Blue:0.3	1
4	Red:0.5, Green:0.3, Blue:0.2	2
5	Red:0.7, Green:0.2, Blue:0.1	2
6	Red:0.3, Green:0.6, Blue:0.1	1
7	Red: 0.5, Green: 0.1,Blue: 0.4	2
8	Red: 0.5, Green: 0.2, Blue: 0.3	2
...	...	...

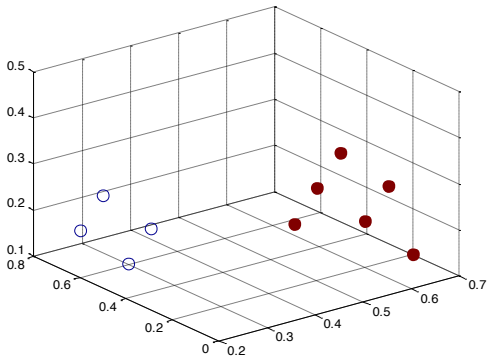


Fig. 1. Mapping vector valued pdfs into Euclidean points

- We develop the new pre-binning technique to discrete the pdfs, which significantly improves the computational and space-efficiency.
- The experimental results in real world data streams prove the effectiveness and efficiency of our approach.

II. PROBLEM STATEMENT

An categorical uncertain attribute, denoted by A^U, is represented by its discrete pdf as $\langle a_1 : p_1, a_2 : p_2, , a_n : p_n \rangle$, where $\{a_1, a_2, , a_n\}$ is the set of all possible values of attribute A^U, and p_i is the probability that $A^U = a_i$. Table I shows an example of uncertain categorical data stream. Here *Color* is the uncertain attribute which has three possible values as {*Red*,*Green* and *Blue*}. An uncertain instance is then represented by a vector valued pdf $\{Red : P_r, Green : P_g, Blue : P_b\}$.

We map a vector valued pdf $\langle a_1 : p_1, a_2 : p_2, , a_d : p_d \rangle$ of an uncertain categorical attribute A^U to a point in the d-dimensional Euclidean space whose coordinate is $\langle p_1, p_2, \ldots, p_d \rangle$. Fig. 1 shows the data points corresponding to uncertain instances in Table reftb:exmp. The distance between two data points measures the similarity of their corresponding vector valued pdfs.

Mapping discrete pdfs to data points helps to manipulate categorical data uncertainty conveniently. First, we transfer probabilistic data instances to fixed points, which enables the directly use of traditional data mining techniques. For example, we can adopt a multi-variable kernel density estimator to estimate the density distribution of pdfs. Second, we can directly use pdfs as the input, instead of the probabilistic attribute values. This property helps train a classification model to reveal the relations between pdfs and class labels. Third, we are now able to obtain an intuitive understanding of the data model in uncertain data.

We incorporate the new uncertainty management into naïve Bayesian classification model to design a classifier for uncertain data streams. The naïve Bayesian classification model is shown in Equation (1).

$$P(c_i|X) = \frac{\prod_{j=1}^{n} P(x_j|C_i)}{P(X)} * P(C_i) \quad (1)$$

Where, $X = \langle x_1, x_2, \cdots, x_n \rangle$ is a test instance, C_i is a class label. Here x_j is a categorical uncertain attribute. The posterior probability $P(C_i|X)$ indicates the membership of X in the class C_i, and X is assigned to the class with maximal membership. We usually use Equation (2) instead of Equation (1) in classification, because P(X) is a constant.

$$P(C_i|X) = \prod_{j=1}^{n} P(x_j|C_i) * P(C_i) \quad (2)$$

In traditional certain databases, the likelihood $P(X|C_i)$ is estimated by the frequency of event X in class C_i. However, we cannot count the frequency of probabilistic attribute values in uncertain data. Therefore, by mapping vector valued pdfs to Euclidean points, we cab measure $P(X|C_i)$ by the density at point X, and estimate it from uncertain data instances in class C_i.

A naïve approach is to use the mean m_i of all data points in C_i to represent the overall density distribution in class C_i so that $P(X|C_i)$ is measured by the distance between m_i and X. A more sophisticated approach is to estimate the density distribution from data, and the density at X is used to measure $P(X|C_i)$. Data stream usually requires one-pass scan in building classification models. And it is very important to train the model with bounded memory and computation cost in uncertain data stream classification.

III. SOLUTION

In this section, we propose two approaches to induce naïve Bayesian classifier in categorical uncertain data streams.

A. A Distance Based Approach

As attributes are assumed to be independent in naïve Bayesian model, we first analyze the computation of posterior probability for one attribute. In uncertain case, a straightforward extension to the traditional approach is to define a new point P_i to represent the data distribution in the class C_i, and calculate $P(X|C_i)$ by the distance between X and P_i.

Suppose A^U is an uncertain categorical attribute with d possible values, we define point P_i as the closest point to all the n observed points in C_i. Let $P_i = (P_{i_1}, P_{i_2}, \ldots, P_{i_d})$, then it minimizes the d-dimensional Euclidean distance in Equation (3).

$$\arg\min_{P_i} \sum_{j=1}^{n} \sqrt{(p_{j_1} - P_{i_1})^2 + \cdots + (p_{j_d} - P_{i_d})^2} \quad (3)$$

Where, $p_j = (p_{j_1}, p_{j_2}, \cdots, p_{j_d})$ are the n data points in C_i. By solving Equation (3), the coordinate of P_i is computed in Equation (4).

$$P_i = \frac{\sum_{j=1}^{n} p_j}{n} \tag{4}$$

Where n is the number of instances in C_i. We can see that P_i is the mean of data observations in C_i, the point P_i is also called the *center* of the class C_i. The uncertain attribute $(a_1 : p_1, a_2 : p_2, \ldots, a_d : p_d)$ of the test instance t is mapped to the data point $p_t = (p_1, p_2, \ldots, p_d)$. The distance between p_t and P_i reflects the membership of p_t belonging to class C_i. Here we use *dot product* to measure the similarity between two discrete pdfs [6], and then $P(X|C_i)$ is computed by Equation (5).

$$P(X|C_i) = X \cdot P_i \tag{5}$$

In data stream classification, when a new element $p_{(n+1)}$ comes, the position of P_i for each categorical uncertain attribute can be updated incrementally by Equation (6).

$$\begin{aligned} P_i^{n+1} &= \frac{\sum_{i=1}^{n+1} p_i}{n+1} = \frac{\frac{n}{n+1} * \sum_{i=1}^{n} p_i}{n} + \frac{p_{(n+1)}}{n+1} \\ &= \frac{n}{n+1} P_i^n + \frac{p_{(n+1)}}{n+1} \end{aligned} \tag{6}$$

We substitute Equation (6) into Equation (2) to induce our distance-based naïve Bayesian classifier for uncertain data streams.

The distance-based approach is simple and fast. We only need to maintain a center point for each attribute in every class, and it cost no additional memory to handle endless incoming data streams. The trade-off of this simplicity is that we assume the density distribution has only one mode, which locates at the center point. However, in most cases, the density distribution is much more complex, and cannot be represented by a single point. Meanwhile, the center point position is sensitive to outliers.

B. A Density Based Approach

In this section, we introduce a density-based approach to induce naïve Bayesian classifier in uncertain streams. In each class, we employ a multi-variable kernel density estimator to estimate the density distribution, which is shown in Equation (8).

$$\hat{f}_h(\boldsymbol{x}) = \frac{1}{n}\frac{1}{h}\sum_{i=1}^{n} K(\frac{x_1 - X_{i_1}}{h_1}, \ldots, \frac{x_d - X_{i_d}}{h_d}) \tag{7}$$

Where, $\boldsymbol{x} = (x_1, x_2, \ldots, x_d)$ is a data point in the d-dimensional space, $X_i = (X_{i_1}, X_{i_2}, \ldots, X_{i_d})$ are n training points, $h = (h_1, h_2, \ldots, h_d)^T$ is the bandwidth matrix and $K(\mu) = K(\mu_1, , \mu_d)$ is the multidimensional kernel function. $K(\mu)$is usually approximated by the multiplicative kernel[14], as shown in Equation (8).

$$K(\mu) = \prod_{i=1}^{d} k(\mu_i) \tag{8}$$

Where k is a uni-variable kernel function, and then the density at $x = (x_1, \ldots, x_d)$ in Equation (8) can be approximated by Equation (9).

$$\hat{f}_h(x_1, \ldots, x_d) = \frac{1}{n}\sum_{i=1}^{n}\prod_{j=1}^{d} k(\frac{x_j - X_{i_j}}{h_j}) \tag{9}$$

Here, k is the 1-dimensional kernel function, and the bandwidth h_j is estimated from the observations in each dimension. However, the kernel density estimator in Equation (9) is not efficient enough for mining data streams, because its computation and memory cost is growing with the incoming data. Therefore, we design a pre-binning approach to improve the efficiency of kernel estimation in data streams.

As the Euclidean data points are mapped from vectored pdfs in our model, the cooperate values of all the data points are bounded in the range of $[0, 1]$. Thus, it is practical and reasonable to discrete them into equal-width bins. For example, if the precision of pdf is measured in unit 0.01, then there is no information lost if we divide the partition [0, 1] into 100 bins equally. By pre-binning the incoming pdfs, we significantly reduce the computation and memory consumption in kernel density estimation. Suppose the range $[0, 1]$ in each dimension of the space is equally divided into k bins, then the entire space is divided into k^d cubes with unit size $1/k$. Here, each cube is represented by its center point. We maintain a kernel table in memory to record historical kernel points. For example, we first discrete the probability values in Table I into five bins, and then insert the pre-binned vectored pdfs into the kernel table, which is shown in Table II.

Here the partition [0,1] is divided into five bins: $b_1 = [0, 0.2]$, $b_2 = (0.2, 0.4]$, $b_3 = (0.4, 0.6]$, $b_4 = (0.6, 0.8]$ and $b_5 = (0.8, 1.0]$. The value of a bin is represented by its center point. For example, the value of b_1 is 0.1. We can see that the data instances with $Id = 7$ and $Id = 8$ in Table I are grouped to one kernel entry in Table II, because they are identical after binning. The size of the kernel table is constant with the number of incoming data, which is at most 5^3 in this example.

The kernel table is updated, when comes a new training instance. Suppose A_t^U is a pre-binned uncertain attribute belonging to class C_i, then if there exists an entry in the kernel table which is identical with A_t^U, we increase the count of that kernel in class C_i by one; otherwise, we add a new entry of the kenerl A_t^U, and initialize its number in C_i as 1.

Now we can revise the kernel density estimator in Equation (10) by using the pre-binning technique.

$$\hat{f}_h(\boldsymbol{x}) = \frac{1}{N}\sum_{i=1}^{S} n_i * \prod_{j=1}^{d}\frac{1}{h_j} k(\frac{x_j - B_{i_j}}{h_j}) \tag{10}$$

Where N is the number of training instances, S is the number of entries in kernel table, and n_i is the number of

TABLE II. AN EXAMPLE OF KERNEL TABLE

Red	Green	Blue	Class Count (N_{C_1}, N_{C_2})
0.1	0.5	0.1	$(1, 0)$
0.5	0.1	0.1	$(0, 1)$
0.1	0.5	0.3	$(1, 0)$
0.5	0.3	0.3	$(0, 1)$
0.7	0.1	0.1	$(0, 1)$
0.3	0.5	0.1	$(1, 1)$
0.5	0.1	0.3	$(0, 2)$
0.5	0.1	0.5	$(0, 1)$

i^{th} kernel in the kernel table in this class. d is the number of possible values for the attribute, h_j is the bandwidth of the j^{th} kernel function, and B_{i_j} is the pre-binned value of i^{th} kernel in j^{th} dimension.

The bandwidth h_j can also be estimated from the kernel table by the plug-in method in Equation (11).

$$h_j = 0.9 * A * N^{-\frac{1}{5}} \tag{11}$$

Where we have

$$A = min(\frac{IQR}{1.34}, \sigma)$$

Here IQR is the inter-quartile and σ is the standard deviation in one dimension. Suppose there are totally N_j entries in j^{th} dimension and each bin has n_i duplicated kernels, then the mean μ_j and standard deviation σ can be estimated by Equation (12).

$$\begin{aligned} \mu_j &= \frac{\sum_{i=1}^{N_j} n_i * b_{ij}}{\sum_{i=1}^{N_j} n_i} \\ \sigma^2 &= \frac{1}{\sum_{i=1}^{N_j} n_i - 1} * \sum_{i=1}^{N_j} (b_{ij} - \mu_j)^2 \end{aligned} \tag{12}$$

Algorithm 1 shows the method to estimate parameter σ in one class C from kernel table T. We first estimate the mean μ by one scan of the entries in the kernel table, and then use μ to compute the standard deviation σ.

Similarly, we can estimate $IQR = Q_3 - Q_1$ from the kernel table. Q_3 is the 75% percentile quartile so that 75% of the values are smaller than Q_3; and Q_1 is the 25% percentile quartile. In each Euclidean dimension of an uncertain attribute, we first select out the entries belonging to class C and sort it by the entries' values. Then, we can directly select the Q_3 and Q by one scanning of the sorted entries. The details of IQR estimation are shown in Algorithm 1.

Now we can incrementally estimate the density distribution in uncertain data streams. When new training data come, we update the kernel table to estimate the density and compute $P(X|C_i)$, which is used to calculate the membership of class C_i for any test point X. X is classified to the class with the maximal membership. Algorithm 3 shows our density based algorithm to induce naïve Bayesian classifier in categorical uncertain data streams.

ALGORITHM 1: Estimating σ in class C from kernel table

Input: T: kernel Table, N: number of entries in T
Output: σ
$\boldsymbol{\mu} \leftarrow 0$
$s \leftarrow 0$
foreach *kernel entry* $\boldsymbol{e} \in T$ **do**
  if $\boldsymbol{e}.N_c > 0$ **then**
    $\boldsymbol{e.k} = \langle \boldsymbol{e}.k_1, \ldots, \boldsymbol{e}.k_d \rangle$
    $\boldsymbol{\mu} = (\boldsymbol{\mu} * s + \boldsymbol{e.k} * e.N_c)/(s + \boldsymbol{e}.N_c)$
    $s = s + \boldsymbol{e}.N_c$
  end
end
$\boldsymbol{\sigma}^2 \leftarrow 0$
foreach *kernel entry* $\boldsymbol{e} \in T$ **do**
  if $\boldsymbol{e.N_c} > 0$ **then**
    $\boldsymbol{e.k} = \langle \boldsymbol{e}.k_1, \ldots, \boldsymbol{e}.k_d \rangle$
    $\boldsymbol{\sigma}^2 = \boldsymbol{\sigma}^2 + (\boldsymbol{e.k} - \boldsymbol{\mu})^2$
  end
end
return $\sqrt{\boldsymbol{\sigma}^2/(s-1)}$

ALGORITHM 2: Estimating IQR in class C from kernel table

Input: T: kernel Table, N: number of entries in T
Output: IQR
$q_1 \leftarrow 0.25 * N$
$q_3 \leftarrow 0.75 * N$
foreach *dimension* d_i **do**
  $v = \phi$
  foreach *kernel entry* $\boldsymbol{e} \in T$ **do**
    if $\boldsymbol{e}.N_C > 0$ **then**
      $v = v \cup \langle \boldsymbol{e}.k_i, \boldsymbol{e}.N_C \rangle$
    end
  end
  sort(v) by $v.k_i$
  $i \leftarrow 0$, count $\leftarrow 0$
  while $i < v.size$ **do**
    count $=$ count $+ v.N_C$
    if *count* $\geq q_1$ **then**
      $Q_1 = v.k_i$
    end
    if *count* $\geq q_3$ **then**
      $Q_3 = v.k_i$
      break;
    end
    $i = i + 1$
  end
  $IQR_i = Q_3 - Q_1$
end
return $IQR = \langle IQR_1, \ldots, IQR_d \rangle$

IV. EXPERIMENTS

A. Setup

We use five real datasets, which are listed in Table III, from UCI repository in our experiments. For all datasets except LED, we add synthetic data uncertainty to raw data by the approach in [8]. Suppose a categorical attribute A in original dataset D has n possible values, then the generated uncertain attribute A^U has the discrete pdf $\langle a_1 : p_1, a_2 : p_2, \ldots, a_n : p_n \rangle$. The original attribute value a_k is defined as the *Main Value*, which is associated with the probability p_m. Here p_m is drawn from a normal distribution $N \sim (\mu, 0.1)$, where μ is a parameter to control the uncertain level and its value is selected randomly from $\{0.6, 0.7, 0.8\}$. All other possible values of A^U

ALGORITHM 3: The density based naïve Bayesian classifier for categorical uncertain streams

Input: A chunk of buffered data $D = D_1 \cup D_2$
D_1:training set, D_2: testing set
Output: predicted labels of D_2
$B \leftarrow$ binning D_1 into k bins
update Kernel table T
estimate parameters σ and IQR to select bandwidth matrix h
$L \leftarrow \phi$
foreach *instance* $t \in D_2$ **do**
  foreach *attribute* $a^u \in t$, $a^u = \{a_1 : p_1, \ldots, a_n : p_n\}$ **do**
    $t \leftarrow \text{bin}(p_1, \ldots, p_n)$
    foreach *class* C_i **do**
      estimate $P(C_i|t.a^u)$ by the density $P(t.a^u|C_i)$
    end
  end
  compute the posterior probability $P(C|t) = \prod_{a^u} P(C_i|t.a^u)$
  $t.C = \arg\max_C P(C|t)$
  $L = L \cup \{t.C\}$
end
return L

TABLE III. DATASETS USED IN EXPERIMENTS

DataSet	# of instance	# of categorical attribute	missing values
Credit	1000	13	N/A
Chess	3196	36	N/A
Voting	435	16	Yes
Mushroom	8124	22	Yes
Led	1 000 000	7	N/A

except the Main value a_k are assigned probabilities p_i so that they satisfy the constraint $\sum_{i=1}^{n} p_i = 1$. For missing attribute values, it is reasonable to assign an equal probability to every possible value in its discrete pdf. Meanwhile, we construct a noisy dataset for comparison purpose by drawing one sample from each pdf-represented uncertain attribute value.

The original dataset LED is inherently imprecise [9]. Thus, we generate the uncertain data stream by aggregating the original data. Every 100 instances of the original dataset are grouped as one uncertain instance. and the probabilities of the uncertain attributes are measured by their frequencies. For example, suppose one attribute A in class C_i has two possible values a_1, a_2 and there are t instances in class C_i among all 100 instances. If the frequency of a_1 in these t instances is x, and the frequency of a_2 is y, then we estimate the probability distribution of A as $\{a_1 : x/t, a_2 : y/t\}$. An uncertain instance is generated by repeating this process in all attributes.

We repeatedly load the categorical uncertain datasets to simulate the uncertain data streams. In our experiment system, an input buffer is used to save 200 incoming data instances. The buffered data are equally divided into four folds. one is randomly selected for testing; while other folds are used as new training data samples.

We implement both the distance-based and the density-based classifiers in our experiments of classifying categorical uncertain data streams. We also implement a batch-mode classical naïve Bayesian classifier in the sampled noisy datasets, for the purpose of comparison. For each dataset except LED, we set different uncertain levels by varying the value of μ. In density-based approach, we select different number of bins from $\{20, 50, 100\}$ in pre-binning process. We analyze the performances of our algorithms to verify the following advantages: (1) prediction accuracy; (2) effect of pre-binning technique; (3) memory efficiency.

B. Results

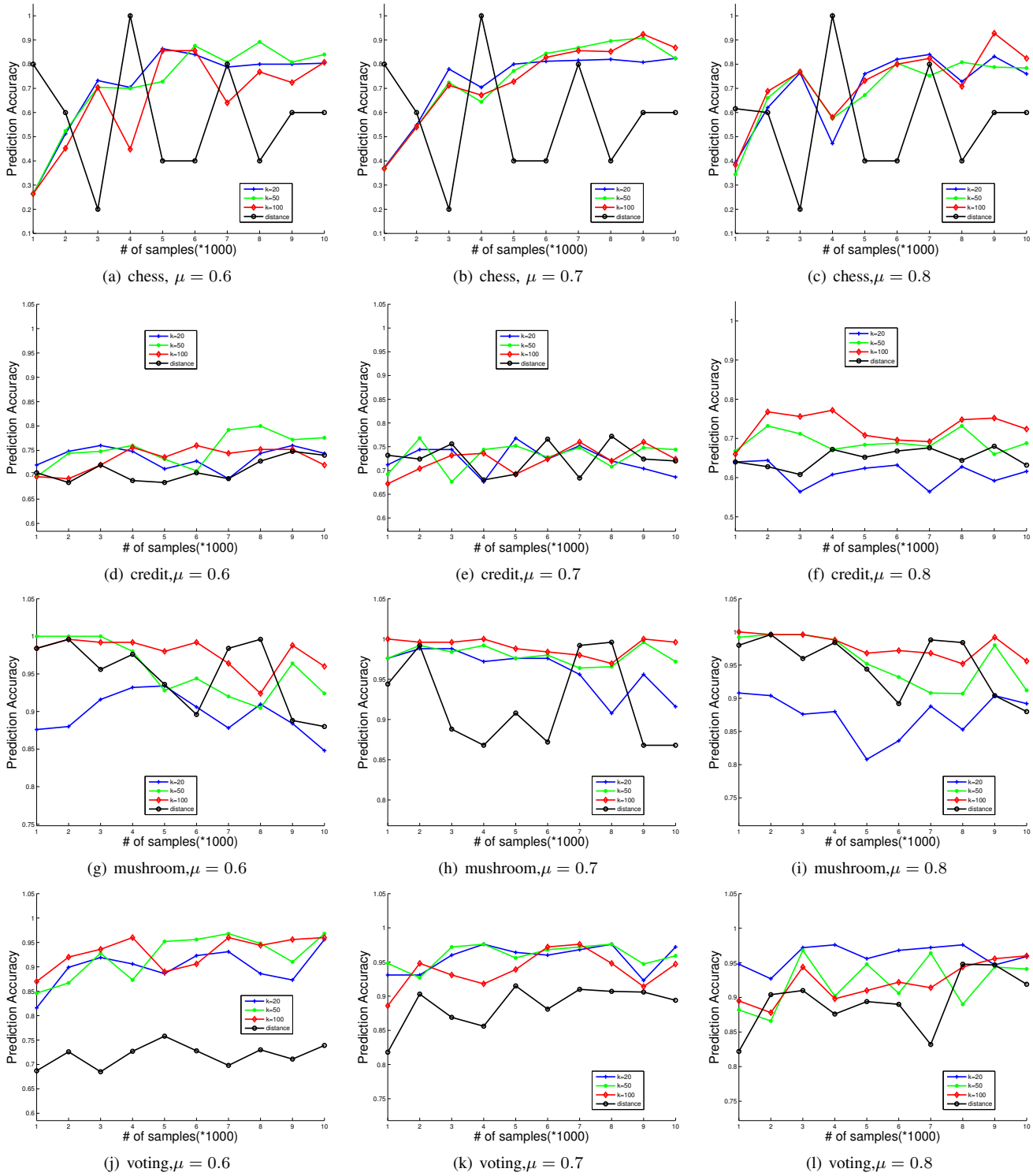
Table IV compares the averaged prediction accuracies in the five uncertain data streams. Our density-based naïve Bayesian classifier outperforms other two methods in most cases. The prediction accuracy is improved when we select a proper value of k, which is the number of bins in pre-binning technique. k corresponds to the round-off error in discretization, and reflects the trade-off between efficiency and accuracy. For example, in data stream *mushroom*, the density-based classifier has the highest accuracy when $k = 100$. The resolution is higher when k is set to a large value, which helps distinguish clustered Euclidean points. This also explains the reason why the density-based classifier has the slightly lower accuracy in *credit* and *mushroom* than other two approaches, when $k = 20$. However, in the data stream *chess*, it achieves the best performance when $k = 20$. The reason is that the smaller k value helps reduce the influence of outliers.

The original LED dataset contains inherent error, and its optimal classification accuracy is proved to be 74% [9]. However, by pre-aggregating the original LED dataset into the categorical uncertain data stream, our density-based classifier achieves the accuracy $95.6\% \sim 99.6\%$, which outperforms the traditional naïve Bayesian classifier and the distance-based approach. This proves that our uncertain management of mapping pdfs into Euclidean points is effective in classifying uncertain data streams.

By comparing to the performance of traditional naïve Bayesian algorithm in Table IV, we can see that our density-based classifier is robust to data uncertainty. The accuracy of traditional naïve Bayesian classifier drops dramatically with the increment of noise; while our density-based approach has relatively similar performance under different uncertain levels.

Fig. 2 compares the performance between our density-based classifier and distance-based algorithm under different uncertain levels in the four data streams. Here each marked point is the prediction accuracy in one iteration, which processes five frames of buffed data. We can see that the density-based approach has a more smooth performance than the distance-based method. The first reason is that the distance-based approach is more sensitive to the influence of outliers. Second, the distance-based approach can only work in the single mode data streams; while the density-based method does not have this constraint and can effectively classify data streams in arbitrary shapes with accumulative training instances. This is also the reason why the density based classifier usually has smaller standard deviation of its prediction accuracy.

Fig. 3 shows the kernel table size in classifying data stream LED. The data stream contains 200 000 uncertain instances in total. In Fig. 3, the kernel table size quickly arrives to almost a constant in every setting of the bin number. Because of the pre-binning technique, we consume bounded memory to build the kernel table for the incoming data. In practice, if we set



(a) chess, $\mu = 0.6$ (b) chess, $\mu = 0.7$ (c) chess,$\mu = 0.8$

(d) credit,$\mu = 0.6$ (e) credit,$\mu = 0.7$ (f) credit,$\mu = 0.8$

(g) mushroom,$\mu = 0.6$ (h) mushroom,$\mu = 0.7$ (i) mushroom,$\mu = 0.8$

(j) voting,$\mu = 0.6$ (k) voting,$\mu = 0.7$ (l) voting,$\mu = 0.8$

Fig. 2. Comparing prediction accuracy under different uncertain levels in four data streams

$k = 100$, then it only needs less than 1MB memory to classify the uncertain stream.

V. Related Works

Recently, many classical data mining algorithms are revised for uncertain data [1], [3], [10], [11]. Specifically, a lot of classifiers are extended to the uncertain versions such as uncertain decision tree classifier [5], uncertain SVM classifier [7] and uncertain rule-based classifier [4]. A naïve Bayesian classifier for uncertain data is proposed in [2], which represents the data uncertainty by continues random variable in either sample-based or formula-based probability distribution. It uses the expectation of the random variables to handle data

TABLE IV. CLASSIFICATION ACCURACY IN FOUR UNCERTAIN DATA STREAMS

Data streams	μ	Density based			Distance based	Traditional NB
		$k = 20$	$k = 50$	$k = 100$		
chess	0.6	0.806 ± 0.017	0.845 ± 0.038	0.760 ± 0.072	0.58 ± 0.220	0.597
	0.7	0.816 ± 0.006	0.868 ± 0.031	0.866 ± 0.032	0.60 ± 0.140	0.551
	0.8	0.796 ± 0.044	0.787 ± 0.019	0.817 ± 0.070	0.56 ± 0.150	0.482
credit	0.6	0.733 ± 0.023	0.770 ± 0.032	0.746 ± 0.014	0.723 ± 0.022	0.682
	0.7	0.711 ± 0.032	0.735 ± 0.015	0.738 ± 0.018	0.701 ± 0.040	0.573
	0.8	$0.606 \pm 0.025^*$	0.689 ± 0.023	0.722 ± 0.025	0.660 ± 0.018	0.431
mushroom	0.6	0.902 ± 0.019	0.932 ± 0.020	0.969 ± 0.024	0.910 ± 0.044	0.743
	0.7	0.954 ± 0.024	0.976 ± 0.011	0.984 ± 0.009	0.927 ± 0.056	0.695
	0.8	$0.874 \pm 0.026^*$	0.927 ± 0.028	0.968 ± 0.014	0.926 ± 0.046	0.477
voting	0.6	0.914 ± 0.030	0.950 ± 0.021	0.945 ± 0.020	0.721 ± 0.014	0.773
	0.7	0.959 ± 0.019	0.964 ± 0.010	0.951 ± 0.022	0.899 ± 0.010	0.682
	0.8	0.964 ± 0.010	0.929 ± 0.026	0.939 ± 0.018	0.907 ± 0.043	0.429
LED	-	0.956 ± 0.027	0.995 ± 0.011	0.996 ± 0.018	0.901 ± 0.051	0.601

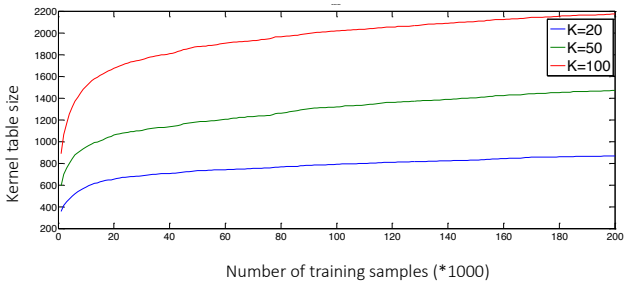


Fig. 3. Kernel Table size in classifying LED data stream

uncertainty, which has the inherent weakness. The approach of mapping vector-valued pdf to data points in multi-dimensional Euclidean space is previously used in indexing uncertain data [12]. In our algorithms, we adopt this mapping to construct more sophisticated classification model to reveal relationship between pdfs and class labels.

Many uncertain data stream mining clustering algorithms are devised in recent years [13], [14]. And [15] introduced a new Gaussian mixture model for processing uncertain data streams. Though data streams classification has been well studied [8], [16], the uncertain data streams bring the classification problem back to the front, and our algorithms are propose to solve this new problem.

VI. CONCLUSION

In this paper, we propose a new approach to construct naïve Bayesian classifier for uncertain categorical data streams. We map the vectored pdfs of uncertain categorical attribute into points in the multi-dimensional Euclidean space to estimate the distribution of pdf inputs,which is used to induce naïve Bayesian classifier. We pre-bin the discrete pdf to guarantee the bounded computation and memory efficiency in classifying uncertain data streams.Experiments proved the outstanding performance of our methods. In the future, we will continue to develop data mining applications in uncertain stream minings.

REFERENCES

[1] W. K. Ngai, B. Kao, C. K. Chui, R. Cheng, M. Chau, and K. Y. Yip, "Efficient clustering of uncertain data," in *Proceedings of the Sixth International Conference on Data Mining*, ser. ICDM '06. Washington, DC, USA: IEEE Computer Society, 2006, pp. 436–445.

[2] J. Ren, S. D. Lee, X. Chen, B. Kao, R. Cheng, and D. Cheung, "Naive bayes classification of uncertain data," in *Proceedings of the 2009 Ninth IEEE International Conference on Data Mining*, ser. ICDM '09. Washington, DC, USA: IEEE Computer Society, 2009, pp. 944–949.

[3] T. Bernecker, H.-P. Kriegel, M. Renz, F. Verhein, and A. Zuefle, "Probabilistic frequent itemset mining in uncertain databases," in *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*, ser. KDD '09. New York, NY, USA: ACM, 2009, pp. 119–128.

[4] C. Gao and J. Wang, "Direct mining of discriminative patterns for classifying uncertain data," in *ACM SIGKDD*, 2010.

[5] B. Qin, Y. Xia, and F. Li, "Dtu: A decision tree for uncertain data," in *Proceedings of the 13th Pacific-Asia Conference on Advances in Knowledge Discovery and Data Mining*, ser. PAKDD '09. Berlin, Heidelberg: Springer-Verlag, 2009, pp. 4–15.

[6] F. Berzal, J.-C. Cubero, N. Marín, and D. Sánchez, "Building multi-way decision trees with numerical attributes," *Inf. Sci.*, vol. 165, no. 1-2, pp. 73–90, Sep. 2004.

[7] J. Bi and T. Zhang, "Support vector machine with input data uncertainty," in *Proceedings of Advances in Neural Information Processing Systems*, 2004.

[8] C. Aggarwal, J. Han, J. Wang, and P. S. Yu, "on demand classification of data streams," in *ACM SIGKDD*, 2004.

[9] M. Tan and L. Eshelman, "Using weighted networks to represent classification knowledge in noisy domains," in *ICML*, 1998.

[10] C. C. Aggarwal and P. S. Yu, "A survey of uncertain data algorithms and applications," *IEEE Trans. on Knowl. and Data Eng.*, vol. 21, no. 5, pp. 609–623, May 2009.

[11] C. Aggarwal, "On density based transforms for uncertain data mining," in *Data Engineering, 2007. ICDE 2007. IEEE 23rd International Conference on*, 2007, pp. 866–875.

[12] S. Singh, C. Mayfield, S. Prabhakar, R. Shah, and S. Hambrusch, "Indexing uncertain categorical data," in *In Proceedings of the 23rd IEEE International Conference on Data Engineering*, 2007, pp. 616–625.

[13] C. Aggarwal, "On high dimensional projected clustering of uncertain data streams," in *ICDE*, 2009.

[14] C. C. Aggarwal and P. S. Yu, "A framework for clustering uncertain data streams," in *Proceedings of the 2008 IEEE 24th International Conference on Data Engineering*, ser. ICDE '08. Washington, DC, USA: IEEE Computer Society, 2008, pp. 150–159.

[15] T. T.L, L. Peng, B. Li, Y. Diao, and L. Anna, "Pods: A new model and processing algorithm for uncertain data streams," in *ACM SIGMOD*, 2010.

[16] H. Wang, W. Fan, and P. S. Yu, "Mining concept-drifting data stream using ensemble classifiers," in *ACM SIGKDD*, 2003.