https://doi.org/10.1016/j.comcom.2021.06.002

RPC: Representative Possible World based Consistent Clustering Algorithm for Uncertain Data

Han Liu[a,b], Xiaotong Zhang[a,b], Xianchao Zhang[a,b], Qimai Li[c], Xiao-Ming Wu[c]

[a] *Dalian University of Technology, School of Software, China*
[b] *Key Laboratory for Ubiquitous Network and Service Software of Liaoning Province, China*
[c] *The Hong Kong Polytechnic University, Department of Computing*

Abstract

Clustering uncertain data is an essential task in data mining and machine learning. Possible world based algorithms seem promising for clustering uncertain data. However, there are two issues in existing possible world based algorithms: (1) They rely on all the possible worlds and treat them equally, but some marginal possible worlds may cause negative effects. (2) They do not well utilize the consistency among possible worlds, since they conduct clustering or construct the affinity matrix on each possible world independently. In this paper, we propose a representative possible world based consistent clustering (RPC) algorithm for uncertain data. First, by introducing representative loss and using Jensen-Shannon divergence as the distribution measure, we design a heuristic strategy for the selection of representative possible worlds, thus avoiding the negative effects caused by marginal possible worlds. Second, we integrate a consistency learning procedure into spectral clustering to deal with the representative possible worlds synergistically, thus utilizing the consistency to achieve better performance. Experimental results show that our proposed algorithm outperforms existing algorithms in effectiveness and performs competitively in efficiency.

Keywords: Uncertain data, clustering, possible world, consistency learning.

☆A preliminary idea of this paper was presented in proceedings of the 28th International Joint Conference on Artificial Intelligence Workshops (IJCAI 2019 Workshops).

Email addresses: liu.han.dut@gmail.com (Han Liu), zxt.dut@hotmail.com (Xiaotong Zhang), xczhang@dlut.edu.cn (Xianchao Zhang), csqmli@comp.polyu.edu.hk (Qimai Li), csxmwu@comp.polyu.edu.hk (Xiao-Ming Wu)

1. Introduction

Clustering aims to automatically discover reasonable partitions for a collection of objects, which is a fundamental task in machine learning and data mining [1]. Most existing clustering algorithms focus on certain data. Due to various reasons like imprecision in physical measurement, randomness in data collection and transmission errors [2, 3, 4, 5], uncertain data is ubiquitous in many real applications, such as sensor networks, biomedical measurement, location tracking, finance and market data analysis, meteorological forecasting and so on [6, 7, 8, 9]. Uncertain data has posed serious challenges to existing clustering algorithms.

Several algorithms have been proposed for clustering uncertain data. Partition-based algorithms, e.g., UK-means [10] and UK-medoids [11], extend traditional k-means or k-medoids to deal with uncertain data by use of expected distance or uncertain distance. However, they reduce complex probability distributions to a single probability distribution or a determinate value, thus cannot handle the uncertain information well [3]. Density-based algorithms, e.g., FDBSCAN [12] and FOPTICS [13], extend traditional DBSCAN [14] or OPTICS [15] for clustering uncertain data by use of probabilistic definitions. However, they suffer from the unreasonable independent distance assumption [16], thus are difficult to obtain satisfactory performance.

Different from partition-based and density-based algorithms, possible world based algorithms, e.g., SC [17] and REP [16], employ multiple independent and identically distributed realizations of an uncertain dataset to deal with data uncertainty, thus reducing the loss of uncertain information and avoiding the independent distance assumption. However, they still have two unaddressed issues: (1) They rely on all the possible worlds and treat them equally, but some marginal possible worlds may cause negative effects on the clustering result. (2) They ignore the consistency among different possible worlds and conduct clustering on each possible world independently. Nevertheless, the consistency is important since different possible worlds can utilize it to transfer useful information for improving the performance.

In this paper, we propose a representative possible world based consistent clustering

(RPC) algorithm for uncertain data, which improves existing algorithms from the following aspects: (1) To alleviate the negative effects caused by marginal possible worlds, we introduce the definition of representative loss, use Jensen-Shannon divergence as the distribution measure, and then design a heuristic strategy for the selection of representative possible worlds. This strategy can be used by any possible world based algorithms to improve the performance. (2) To utilize the consistency to achieve better performance, we integrate a consistency learning procedure into spectral clustering to deal with the representative possible worlds synergistically. Extensive experimental results on real benchmark datasets and real world uncertain datasets demonstrate the superiority of the proposed algorithm over the existing ones.

The preliminary idea of this paper was presented in IJCAI 2019 workshops [18]. To ensure the paper to be more complete and self-contained, we have added a lot of algorithm details, experimental results and comprehensive analysis. The rest of this paper is organized as follows: In section 2 and 3, we review the related work and introduce some preliminary knowledge; In section 4 and 5, we propose our algorithm and show the experimental results; Finally in section 6, we conclude the paper and present the future work.

2. Related Work

2.1. Traditional algorithms

2.1.1. Partition-based algorithms

UK-means [10] is the first partition-based algorithm for clustering uncertain data. It extends the traditional k-means by using expected distance. To improve the efficiency of UK-means, [19, 20, 21, 22] use various pruning techniques to avoid the computation of redundant expected distances. CK-means [23] optimizes UK-means by resorting to the moment of inertia of rigid bodies. DUK-means [24] is an improved version of UK-means, which is specifically designed for distributed network environment. UK-medoids [11] employs uncertain distance to extend the traditional k-medoids. MMVar [25] uses a novel objective function which aims to minimize the variance of cluster mixture models. UCPC [26] introduces the notion of uncertain centroid and it is a local search based heuristic algorithm. All these algorithms can deal with uncertain data to some extent. However, the nature behind them is to reduce complex probability distributions to a single probability distribution or a determinate value, thus they cannot handle the uncertain information well [3].

2.1.2. Density-based algorithms

FDBSCAN [12] and FOPTICS [13] are the first density-based and hierarchical density-based algorithms for clustering uncertain data respectively. They introduce a series of probabilistic definitions like distance density function, core object probability, reachability probability, fuzzy core distance, fuzzy reachability-distance to extend the traditional DBSCAN [14] and OPTICS [15]. Zhang et al. [3] find the limitations of losing uncertain information, high time complexity and nonadaptive threshold in FDBSCAN and FOPTICS, and then propose novel density-based algorithm PDBSCAN and hierarchical density-based algorithm POPTICS for clustering uncertain data. However, these density-based algorithms rely on the unreasonable independent distance assumption [16], thus are difficult to obtain satisfactory clustering results.

2.2. Possible world based algorithms

SC [17] is the first possible world based algorithm for clustering uncertain data. It conducts clustering on each possible world independently and integrates the clustering results into one final result. REP [16] also conducts clustering on each possible world independently, but it selects the representative clustering result as the final result. The demo of REP can be found in [27]. Recently, [6] tries to leverage the consistency principle for clustering uncertain data. It constructs the affinity matrix for each possible world independently and then learns a consensus affinity matrix for clustering uncertain data. However, the consistency learning method introduced in [6] lacks the procedure of updating the affinity matrix of each possible world, thus reducing the ability of consistency learning. Possible world based algorithms avoid the issues in traditional algorithms, thus seem more promising. However, as we point out hereinafter, there are some unaddressed issues in existing possible world based algorithms.

3. Preliminaries

3.1. Consistency Principle

Consistency principle is a common assumption, which has been widely used in machine learning domain, e.g., multi-view learning [28, 29], multiple kernel learning [30], latent space learning [31] and so on. Its definition is as follows [32].

Definition 1. *Consistency principle: Given a dataset which has multiple representations, consistency principle refers to an assumption that the class labels and cluster structures of the multiple representations are consistent.*

By using consistency principle to minimize the disagreement of different representations, we can improve the algorithm performance. The detailed proof can refer to [33]. Here we take a simple example to explain the reason. Given a dataset D which has two representations R^1 and R^2, f^1 and f^2 are the hypotheses of R^1 and R^2 respectively. According to [33], by using consistency principle, we can have the following inequality:

$$P(f^1 \neq f^2) \geqslant \max\{P_{error}(f^1), P_{error}(f^2)\}, \tag{1}$$

where P denotes the probability, and P_{error} denotes the error probability. From the inequality, it can be seen that the probability of the disagreement of two different hypotheses is the upper bound of the error probability of each hypothesis [33]. Therefore, by minimizing the disagreement of different hypotheses, the error probability of each hypothesis will be minimized. If we assume different representations share a common hypothesis, by minimizing the disagreement between each hypothesis and the shared hypothesis, the error probability of the shared hypothesis will also be minimized.

3.2. Uncertain Data and Possible World

Uncertain data can be considered at table, tuple or attribute level [34]. For clustering uncertain data, we mainly focus on attribute level uncertainty. That is to say, each uncertain object is represented as a random variable with a probability distribution, which is associated with the probability that the object appears at any position in a multidimensional space.

Possible world is an effective tool to model uncertain data [35, 34, 36]. Its definition is as follows [36].

Definition 2. *Possible world: Let $UD = \{O_1, O_2, ..., O_n\}$ be an uncertain dataset. A possible world $pw = \{o_1, o_2, ..., o_n\}$ $(o_i \in O_i)$ is a set of instances such that each instance is taken from each corresponding uncertain object. Let PW be the set of all the possible worlds, $P(pw)$ be the existence probability of pw, then $\sum_{pw \in PW} P(pw) = 1$.*

Possible world can be generated through the inversion method. More information and proofs can refer to [37, 38].

3.2.1. Consistency Principle for Possible World

According to the definition of possible world, different possible worlds come from the same uncertain dataset and they are a number of independent and identically distributed realizations of an uncertain dataset [36]. Therefore, if we treat each possible world as one representation of the uncertain dataset, by the concept of consistency principle, we can have the following consistency principle for possible world: *the class labels and cluster structures of different possible worlds are consistent*.

In general, the consistency principle for possible world conforms to the reality well, i.e., in most cases the class labels and cluster structures of different possible worlds are consistent. For example, in Figure 1, O_1, O_2, O_3 are uncertain objects, and o_1^i, o_2^i, o_3^i, o_4^i, o_5^i are the possible instances of O_i $(i \in \{1, 2, 3\})$. If we divide O_1, O_2, O_3 into two clusters, based on the geometric information, O_1 and O_2 should belong to one cluster, and O_3 should belong to the other cluster. For the possible worlds pw_1, pw_2, pw_3, pw_4 with their components shown in Figure 1, it is easy to find that their class labels and cluster structures are consistent.

However, the consistency principle for possible world is not absolute. In some cases, abnormal possible worlds violate the principle, and we call this kind of possible worlds as the marginal ones. Formally, the definition of marginal possible world is as follows.

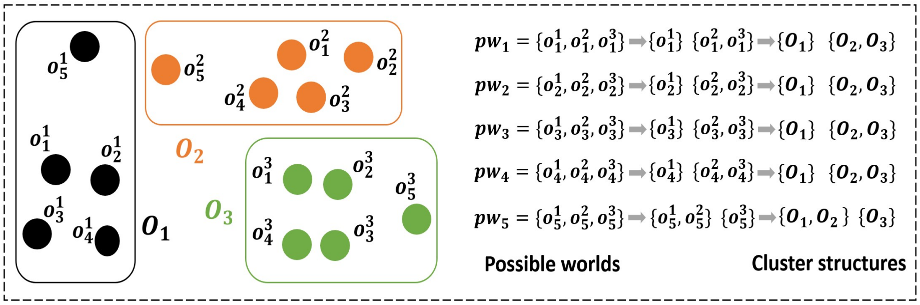

o_5^1
o_1^1
o_2^1
o_3^1
o_4^1
O_1
o_5^2
o_1^2
o_2^2
o_4^2
o_3^2
O_2
O_3
o_1^3
o_2^3
o_5^3
o_4^3
o_3^3
$pw_1 = \{o_1^1, o_1^2, o_1^3\} \rightarrow \{o_1^1\}\ \{o_1^2, o_1^3\} \rightarrow \{O_1\}\ \{O_2, O_3\}$
$pw_2 = \{o_2^1, o_2^2, o_2^3\} \rightarrow \{o_2^1\}\ \{o_2^2, o_2^3\} \rightarrow \{O_1\}\ \{O_2, O_3\}$
$pw_3 = \{o_3^1, o_3^2, o_3^3\} \rightarrow \{o_3^1\}\ \{o_3^2, o_3^3\} \rightarrow \{O_1\}\ \{O_2, O_3\}$
$pw_4 = \{o_4^1, o_4^2, o_4^3\} \rightarrow \{o_4^1\}\ \{o_4^2, o_4^3\} \rightarrow \{O_1\}\ \{O_2, O_3\}$
$pw_5 = \{o_5^1, o_5^2, o_5^3\} \rightarrow \{o_5^1, o_5^2\}\ \{o_5^3\} \rightarrow \{O_1, O_2\}\ \{O_3\}$
Possible worlds
Cluster structures

Figure 1: Consistency principle for possible world.

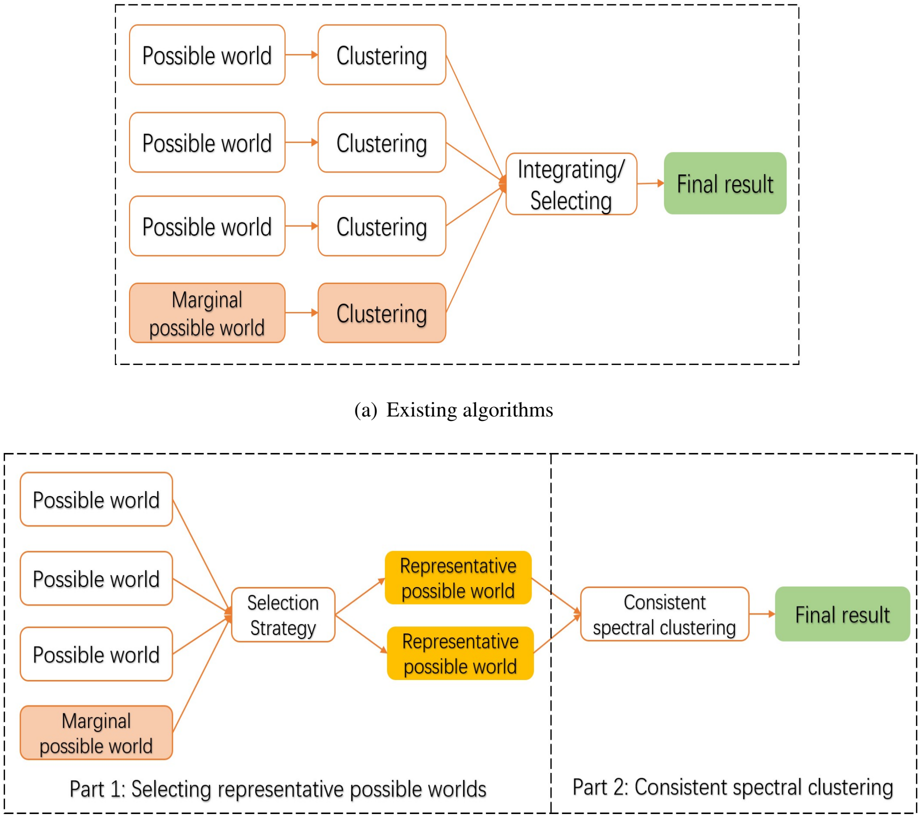
Definition 3. *Marginal possible world: Let PW be the set of all the possible worlds, marginal possible world refers to the possible world whose class label and cluster structure have large differences with most possible worlds in PW.*

For example, in Figure 1, pw_5 is a possible world which consists of some abnormal instances. As the class label and cluster structure of pw_5 are very different from most possible worlds, pw_5 is a marginal possible world.

3.3. Unaddressed Issues

3.3.1. Negative effects caused by marginal possible worlds

Figure 2(a) shows the framework of existing possible world based algorithms. From the framework, it can be seen that existing possible world based algorithms rely on all the possible worlds and treat them equally. However, marginal possible worlds belong to the abnormal ones, their class labels and cluster structures have large differences with most possible worlds, which may disturb the integrating or selecting procedure of existing possible world based algorithms and cause negative effects on the clustering result. To solve this issue, we propose to select some representative possible worlds to filter out marginal possible worlds. By representative possible worlds we mean a subset of all the possible worlds which has a strong ability to represent all the possible worlds. As marginal possible worlds are abnormal and their representative ability is weak, we can filter out marginal possible worlds and avoid the negative effects by selecting representative possible worlds.



(a) Existing algorithms

(b) The proposed algorithm

Figure 2: Frameworks of possible world based algorithms.

3.3.2. Ignoring the consistency principle for possible world

The consistency principle makes it possible to transfer useful information among different possible worlds, which can potentially improve the clustering quality. However, as shown in Figure 2(a), existing possible world based algorithms ignore the consistency principle for possible world and conduct clustering on each possible world independently. To solve this issue, we propose a consistent spectral clustering method which can minimize the disagreement of different possible worlds, thus achieving the consistency learning and improving the clustering performance.

4. The Proposed Algorithm

The proposed algorithm consists of two parts: selecting representative possible worlds and consistent spectral clustering. The framework is shown in Figure 2(b).

Notations: For an uncertain dataset $UD = \{O_1, O_2, ..., O_n\}$ in a d-dimensional independent space, PW denotes the set of all the possible worlds, $PW = \{pw_i|i = 1, 2, ..., M\}$, M is the number of possible worlds in PW. PWR denotes the representative possible world set, $PWR = \{pwr_j|j = 1, 2, ..., R\}$, R is the number of possible worlds in PWR. PWU denotes the unrepresentative possible world set, $PWU = \{pwu_k|k = 1, 2, ..., M - R\}$, $M - R$ is the number of possible worlds in PWU. Here $PW = PWR \cup PWU$.

4.1. Selecting Representative Possible Worlds

By selecting representative possible worlds, we can filter out marginal possible worlds and avoid the waste of time caused by redundant possible worlds. In order to select representative possible worlds, we introduce the definition of representative loss, use Jensen-Shannon divergence as the distribution measure, and then design a heuristic strategy for the selection of representative possible worlds.

4.1.1. Representative Loss

Intuitively, given any two possible worlds pw and pw', if we want to use pw to represent pw', then the smaller the difference between pw and pw', the less the loss that pw represents pw'. We aim to select PWR from PW to represent PW. As $PW = PWR \cup PWU$ and the loss that PWR represents PWR is equal to 0, then the loss that PWR represents PW is equal to the loss that PWR represents PWU. Based on these observations, we have the following definition.

Definition 4. *Representative Loss: Let PWR be the representative possible world set and $pwr_j \in PWR$, PWU be the unrepresentative possible world set and $pwu_k \in PWU$. If using PWR to represent PWU, then the representative loss, denoted by $L(PWR \to PWU)$, can be defined as:*

$$L(PWR \to PWU) = \sum_{k=1}^{M-R} \min_{pwr_j} \Phi(pwr_j, pwu_k), \tag{2}$$

where $\Phi(pwr_j, pwu_k)$ is the difference between pwr_j and pwu_k, $M-R$ is the number of possible worlds in PWU.

From this definition, it can be seen that if we know how to compute the difference between possible worlds, we can get the representative loss that PWR represents PWU, i.e., the representative loss that PWR represents PW.

4.1.2. Jensen-Shannon Divergence between Possible Worlds

As a possible world can be regarded as a probability distribution, we can compute the difference between possible worlds by Jensen-Shannon divergence [39]. Compared with KL divergence [40], Jensen-Shannon divergence is symmetric and finite, therefore it is more suitable as the representative loss measure.

Given any two possible worlds pw and pw', the Jensen-Shannon divergence between them can be defined as:

$$JSD(pw||pw') = \frac{1}{2}D(P_{pw}||H) + \frac{1}{2}D(P_{pw'}||H), \tag{3}$$

where P_{pw} and $P_{pw'}$ are the probability distributions of pw and pw' respectively, and $H = \frac{1}{2}(P_{pw} + P_{pw'})$. $D(P||Q)$ is the KL divergence between two probability distributions P and Q. For continuous probability distributions P and Q with a variable x in a domain $\mathbb{D}$, $D(P||Q)$ is defined as:

$$D(P||Q) = \int_{\mathbb{D}} f(x) log\frac{f(x)}{g(x)}dx, \tag{4}$$

where $f(x)$ and $g(x)$ are the probability density functions of P and Q. According to Eq.(4), $D(P||Q)$ can also be expressed as:

$$D(P||Q) = E(log\frac{f(x)}{g(x)}), \tag{5}$$

where E denotes the expectation. According to the law of large numbers and Eq.(5), given a sample set S, $D(P||Q)$ can be estimated by:

$$D(P||Q) = \frac{1}{|S|}\sum_{x\in S} log\frac{f(x)}{g(x)}, \tag{6}$$

where $|S|$ denotes the number of objects in S.

We employ the kernel density estimation method [41] to obtain the probability density functions f_{pw} and $f_{pw'}$ of the probability distributions P_{pw} and $P_{pw'}$. Specifically, f_{pw} can be estimated as:

$$f_{pw}(x) = \frac{1}{|pw| \prod_{j=1}^{d} h_j} \sum_{o \in pw} \prod_{j=1}^{d} K(\frac{x.D_j - o.D_j}{h_j}). \tag{7}$$

In Eq.(7), o is an object in pw and it can be represented by $(o.D_1, o.D_2, ..., o.D_d)$, d denotes the total dimensionality, and $|pw|$ denotes the number of objects in pw. K denotes the kernel function, and we use the most common Gaussian kernel function. h_j denotes the bandwidth of the j-th dimension, which can control the smoothing level. For Gaussian kernel function, we set $h_j = 1.06 \times \delta_j |pw|^{-\frac{1}{5}}$ according to the Silverman's rule of thumb [41], where δ_j is the standard deviation of the j-th dimension of the objects in pw.

By using Jensen-Shannon divergence as the distribution measure to compute the difference between possible worlds, i.e., replacing $\Phi(pwr_j, pwu_k)$ in Eq.(2) with $JSD(pwr_j || pwu_k)$, we can get the representative loss.

4.1.3. Selection Strategy

Our goal is to select a given number of possible worlds as the representative possible worlds. In general, a good representative possible world set should have a strong representative ability, i.e., its corresponding representative loss should be small. Inspired by this observation, we propose the following selection strategy:

Let PWR be the representative possible world set, and PWU be the unrepresentative possible world set. Now select a possible world pwu^* from PWU and move pwu^* to PWR, if we want the new representative possible world set $PWR \cup pwu^*$ to be the best, then the selection strategy should ensure the representative loss that $PWR \cup pwu^*$ represents $PWU \backslash pwu^*$ to be the minimum. Formally:

$$pwu^* = \underset{pwu^*}{\arg\min}\, L(PWR \cup pwu^* \to PWU \backslash pwu^*). \tag{8}$$

From Eq.(8), it can be seen that pwu^* should have a strong representative ability. Marginal possible worlds belong to the abnormal ones and their representative ability is poor, therefore this selection strategy can filter out marginal possible worlds.

Based on the selection strategy, we design a heuristic method to select the representative possible worlds, which is shown in Algorithm 1 (Part 1). After generating the set of all the possible worlds PW, it initializes the representative possible world set $PWR = \emptyset$ and the unrepresentative possible world set $PWU = PW$, and calculates the JSD between any two possible worlds in PW. Then according to Eq.(8), it selects a possible world pwu^* from PWU, moves it to PWR, and updates $PWR \leftarrow PWR \cup pwu^*$ and $PWU \leftarrow PWU \backslash pwu^*$. Here pwu^* should ensure that the new representative possible world set $PWR \cup pwu^*$ has the minimum representative loss to represent $PWU \backslash pwu^*$. This procedure is repeated until the algorithm obtains the required number of representative possible worlds.

4.2. Consistent Spectral Clustering

We integrate a consistency learning procedure into spectral clustering to deal with the representative possible worlds synergistically.

4.2.1. Spectral Clustering

Assume that pwr_j is a possible world from the representative possible world set PWR and $PWR = \{pwr_j | j = 1, 2, ..., R\}$, where R denotes the number of possible worlds in PWR. $W^{(j)}$ is the similarity matrix of pwr_j, which is computed by the Gaussian kernel. $L^{(j)}$ is the normalized Laplacian matrix of pwr_j and $L^{(j)} = D^{(j)^{-\frac{1}{2}}} W^{(j)} D^{(j)^{-\frac{1}{2}}}$. $D^{(j)}$ is a diagonal matrix and $D^{(j)}(i,i) = \sum\limits_{l=1}^{n} W^{(j)}(i,l)$, where n denotes the number of objects in pwr_j. For the representative possible world pwr_j, the objective function of spectral clustering is:

$$\begin{aligned} &\max_{U^{(j)}} tr(U^{(j)^T} L^{(j)} U^{(j)}), \\ &s.t.\ U^{(j)^T} U^{(j)} = I, \end{aligned} \tag{9}$$

where $tr(\cdot)$ denotes the trace of a matrix. $U^{(j)} \in \mathbb{R}^{n \times k}$ is composed by k eigenvectors corresponding to the k largest eigenvalues of $L^{(j)}$.

4.2.2. Consistency Learning

The eigenvector matrix $U^{(j)}$ can reflect the cluster structure of the representative possible world pwr_j. To meet the requirement of consistency, we assume that each eigenvector matrix $U^{(j)} \in \mathbb{R}^{n \times k}$ tends to a common eigenvector matrix $U^* \in \mathbb{R}^{n \times k}$. Then by minimizing the disagreement between each $U^{(j)}$ and U^*, we can achieve the consistency learning among different possible worlds. For the disagreement between $U^{(j)}$ and U^*, we use the squared Euclidean distance between the similarity matrices to measure it:

$$\begin{aligned} &Dis(U^{(j)}, U^*) = ||S_{U^{(j)}} - S_{U^*}||_F^2, \\ &s.t.\ \ {U^{(j)}}^T U^{(j)} = I,\ \ {U^*}^T U^* = I, \end{aligned} \tag{10}$$

where $S_{U^{(j)}}$ and S_{U^*} denote the similarity matrices of $U^{(j)}$ and U^*, and $||\cdot||_F$ denotes the Frobenius norm of the matrix.

Considering the feasibility of optimization, we use the commonly adopted inner product to compute the similarity matrix, i.e., $S_{U^{(j)}} = U^{(j)} {U^{(j)}}^T$. Then with some manipulations, minimizing Eq.(10) can be transformed as:

$$\begin{aligned} &\max_{U^{(j)}, U^*} tr(U^{(j)} {U^{(j)}}^T U^* {U^*}^T), \\ &s.t.\ \ {U^{(j)}}^T U^{(j)} = I,\ \ {U^*}^T U^* = I. \end{aligned} \tag{11}$$

4.2.3. Overall Objective Function and Optimization

By integrating the objective functions of spectral clustering and consistency learning, we can get the overall objective function of consistent spectral clustering as follows:

$$\begin{aligned} &\max_{U^{(j)}, U^*} \sum_{j=1}^{R} (tr({U^{(j)}}^T L^{(j)} U^{(j)}) + tr(U^{(j)} {U^{(j)}}^T U^* {U^*}^T)), \\ &s.t.\ \ {U^{(j)}}^T U^{(j)} = I,\ \ \forall 1 \leqslant j \leqslant R,\ \ {U^*}^T U^* = I. \end{aligned} \tag{12}$$

For Eq.(12), we can employ the alternative iteration method to solve it.

(1) Optimizing Eq.(12) with respect to U^*. Fix each $U^{(j)}$, then Eq.(12) becomes:

$$\begin{aligned} &\max_{U^*} \sum_{j=1}^{R} tr(U^{(j)} {U^{(j)}}^T U^* {U^*}^T), \\ &s.t.\ \ {U^*}^T U^* = I. \end{aligned} \tag{13}$$

Algorithm 1 RPC

```
Input: Uncertain dataset $UD = \{O_1, O_2, ..., O_n\}$, the number of clusters $k$, the number of all
    the possible worlds $M$, the number of representative possible worlds $R$.
Output: The clusters $C_1, C_2, ..., C_k$.
  Part 1: Selecting representative possible worlds (Lines 1-5)
  1: Generate $PW$, initialize $PWR = \emptyset$ and $PWU = PW$, and calculate the $JSD$ between
  any two possible worlds in $PW$
  2: Repeat
  3:     Select a possible world $pwu^*$ from $PWU$ by Eq.(8)
  4:     $PWR \leftarrow PWR \cup pwu^*$, $PWU \leftarrow PWU \backslash pwu^*$
  5: Until $|PWR| \geqslant R$, $|PWR|$ denotes the current number of possible worlds in $PWR$
  Part 2: Consistent spectral clustering (Lines 6-12)
  6: For $\forall pwr_j \in PWR$, compute $W^{(j)}$, $D^{(j)}$, $L^{(j)}$
  7: For $\forall pwr_j \in PWR$, compute the $k$ eigenvectors corresponding to the $k$ largest
  eigenvalues of $L^{(j)}$ and use them to initialize the corresponding $U^{(j)}$
  8: Repeat
  9:     Update $U^*$ by solving Eq.(14)
  10:    Update each $U^{(j)}$ by solving Eq.(16)
  11: Until Eq.(12) is convergent
  12: Run $k$-means on $U^*$ and get the clusters $C_1, C_2, ..., C_k$
```

Eq.(13) can be written as:

$$\begin{gathered} \max_{U^*} tr({U^*}^T (\sum_{j=1}^{R} U^{(j)} {U^{(j)}}^T) U^*), \\ s.t.\ {U^*}^T U^* = I. \end{gathered} \tag{14}$$

It is easy to find that optimizing Eq.(14) is equivalent to solve the standard spectral clustering with a modified Laplacian matrix $\sum_{j=1}^{R} U^{(j)} {U^{(j)}}^T$, i.e., the solution U^* is composed by k eigenvectors corresponding to the k largest eigenvalues of $\sum_{j=1}^{R} U^{(j)} {U^{(j)}}^T$.

(2) Optimizing Eq.(12) with respect to one of the $U^{(j)}$s. Fix the other $U^{(j)}$s and U^*, then Eq.(12) becomes:

$$\begin{aligned} &\max_{U^{(j)}} tr({U^{(j)}}^T L^{(j)} U^{(j)}) + tr(U^{(j)} {U^{(j)}}^T U^* {U^*}^T), \\ &\qquad s.t.\ {U^{(j)}}^T U^{(j)} = I. \end{aligned} \tag{15}$$

Eq.(15) can be written as:

$$\begin{aligned} &\max_{U^{(j)}} tr({U^{(j)}}^T (L^{(j)} + U^* {U^*}^T) U^{(j)}), \\ &\qquad s.t.\ {U^{(j)}}^T U^{(j)} = I. \end{aligned} \tag{16}$$

Optimizing Eq.(16) is similar with optimizing Eq.(14), therefore the solution $U^{(j)}$ is composed by k eigenvectors corresponding to the k largest eigenvalues of $L^{(j)} + U^* {U^*}^T$. The overall procedure of consistent spectral clustering is shown in Algorithm 1 (Part 2).

5. Experiments

5.1. Datasets

5.1.1. Real benchmark datasets

We conduct experiments on 7 real benchmark datasets. The details of the datasets are shown in Table 1. These datasets are originally established as collections of data with determinate values, we follow the method in [3, 16] to generate uncertainty for these datasets. We generate uncertainty with 3 kinds of distributions: uniform distribution (U), Gaussian distribution (G) and logistic distribution (L).

5.1.2. Real world uncertain datasets

We also conduct experiments on 3 real world uncertain datasets: the movement dataset [1], the NBA dataset [2] and the weather dataset [3].

[1] http://archive.ics.uci.edu/ml/

[2] http://espn.go.com/nba/

[3] http://bcc.ncc-cma.net/

Table 1: Real benchmark datasets.

Dataset	#Objects	#Attributes	#Classes
Wine	178	13	3
Glass	214	9	6
Ecoli	327	7	5
Image	2310	19	7
Libras	360	90	15
USPS	929	256	10
Waveform	5000	21	3

(1) Movement: it consists of 13197 radio signal records about 314 temporal sequences. Each record has four dimensions which are respectively corresponding to four sensor nodes. According to user movement path, the dataset is divided into six classes. Each temporal sequence is treated as an uncertain object and each record of the temporal sequence is treated as a possible value of the uncertain object.

(2) NBA: it consists of 2197 records about the top 300 players in ESPN 2015 rank. Each record has five dimensions: points, rebounds, assists, steals and blocks. According to season average performance, they are divided into three classes: star player, key player and role player. Each player is treated as an uncertain object and each season average performance of the player is treated as a possible value of the uncertain object.

(3) Weather: it consists of 18360 records about 153 stations around China. Each station contains the monthly average weather condition from 2006 to 2015. Each record has two dimensions: average temperature and average precipitation. According to [42], each station is labeled with a climate type. In total, we have three types of climates: temperate continental climate, temperate monsoon climate and tropical/subtropical monsoon climate. The stations with the same label are considered to be in the same class. Each station is treated as an uncertain object and each monthly average weather condition of the station is treated as a possible value of the uncertain object.

5.2. Experimental Setup

5.2.1. Baselines

We compare the RPC algorithm with the state-of-the-art clustering algorithms for uncertain data, including UK-means (UKM), CK-means (CKM), UK-medoids (UKMD), MMVar (MMV), UCPC, FDBSCAN (FDB), FOPTICS (FOP), PDBSCAN (PDB), SC and REP. We also compare with the improved versions of SC and REP, which use our proposed selection strategy to select the representative possible worlds and then perform the original SC and REP on the representative possible worlds, and we call them RP-SC and RP-REP.

5.2.2. Settings

For UK-means, CK-means, UK-medoids, MMVar, UCPC and RPC, the sets of initial centroids or partitions are randomly selected. To avoid that the clustering results are affected by random chance, we average the results over 10 different runs. For FDBSCAN, FOPTICS, PDBSCAN, SC, REP, RP-SC and RP-REP, since these algorithms are sensitive to parameters, we adjust the parameters continuously until the performance of each method becomes the best and stable. The methods of determining the parameters can refer to [3, 12, 13, 16, 17].

5.2.3. Evaluation metrics

We adopt two widely used metrics [1]: clustering accuracy (ACC) and normalized mutual information (NMI) to evaluate the clustering results.

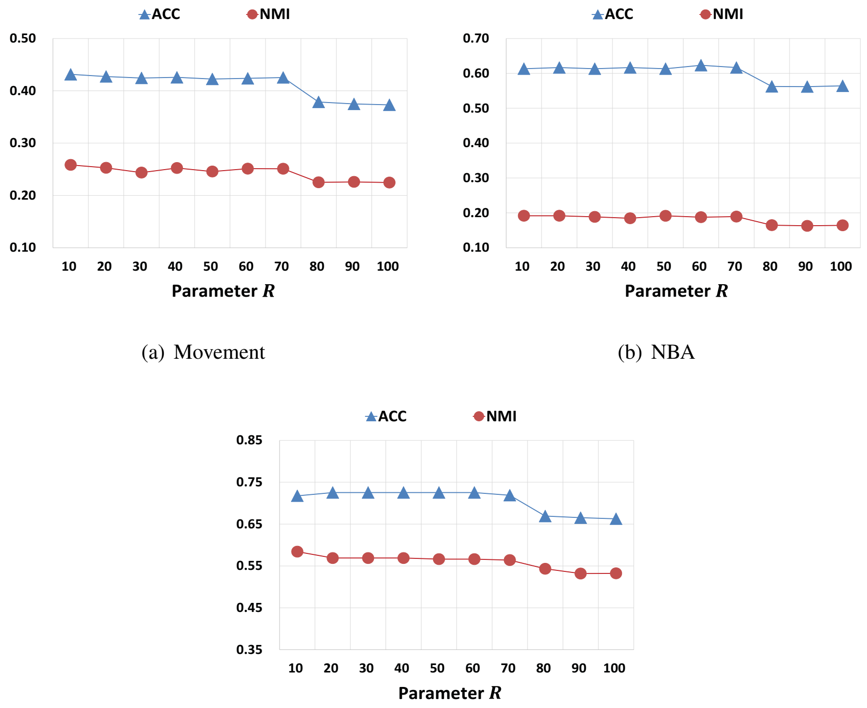
ACC: Given a result, the ACC can be calculated as

$$ACC = \frac{\sum_{i=1}^{n} \chi(r_i, l_i)}{n}, \tag{17}$$

where n denotes the number of objects, r_i denotes the true label of object o_i, l_i denotes the label of object o_i obtained from the algorithm, $\chi(x, y)$ is a logical judgement function that equals 1 if $x = y$ and equals 0 otherwise.

NMI: Given a result, the NMI can be calculated as

$$NMI = \frac{\sum\limits_{c_i \in C, c'_j \in C'} p(c_i, c'_j) \log \frac{p(c_i, c'_j)}{p(c_i) p(c'_j)}}{\sqrt{\sum\limits_{c_i \in C} p(c_i) \log p(c_i) \sum\limits_{c'_j \in C'} p(c'_j) \log p(c'_j)}}, \tag{18}$$



(a) Movement

(b) NBA

(c) Weather

Figure 3: The performance of RPC with different R on real world uncertain datasets.

where C and C' denote the cluster sets from the ground truth and the algorithm respectively, $p(c_i)$ and $p(c'_j)$ denote the probabilities that an object arbitrarily selected from the dataset belongs to the clusters c_i and c'_j respectively, and $p(c_i, c'_j)$ is the joint probability that this arbitrarily selected object belongs to the clusters c_i and c'_j simultaneously.

5.3. *Parameter Investigation for RPC*

(1) For parameter k, we follow the common practice [6, 16] to set k to the true number of classes in the datasets.

(2) For parameter M, the investigation results in previous possible world based methods show that setting $M = 100$ is enough to obtain satisfactory results [16, 17], so we set $M = 100$.

(3) For parameter R, Figure 3 shows the performance of RPC with different R on real world uncertain datasets. From the results, it can be seen that when R is within 10$\sim$70, the clustering performance is always good and stable. When the parameter R is larger than 70, the clustering performance will be affected seriously, which is because that the remaining 30 possible worlds contain many marginal ones. As selecting too many representative possible worlds will result in a waste of time to some extent, in this paper we set $R = 10$ and report the corresponding results.

5.4. Effectiveness

Table 2 and 3 show the effectiveness results. For each algorithm, the last two rows of these tables report: (1) the score averaged over all the datasets and distributions (all avg.ACC/NMI); (2) the overall gain which is computed as the difference between the overall average score of RPC and the overall average scores of other algorithms (all avg.ACC/NMI.gain).

From the overall average scores, it can be seen that RPC performs the best. RP-SC and RP-REP respectively perform better than SC and REP, but not as well as RPC. This is because that compared with SC and REP, RP-SC and RP-REP select the representative possible worlds, thus avoiding the negative effects caused by marginal possible worlds. However, compared with RPC, RP-SC and RP-REP do not make use of the consistency principle among different possible worlds. UK-means, CK-means, UK-medoids, MMVar and UCPC perform worse than RPC. The reason is that these algorithms reduce complex probability distributions to a single probability distribution or a determinate value, which may cause the loss of uncertain information. FDBSCAN, FOPTICS and PDBSCAN also perform worse than RPC. The reason is that they rely on the unreasonable independent distance assumption. All in all, in terms of effectiveness, RPC performs much better than the compared algorithms.

5.5. Efficiency

We report the efficiency results (in milliseconds) on real world uncertain datasets, which is shown in Figure 4. Other datasets have the similar trend. From the results, it can be seen that UK-medoids is the slowest. RPC runs faster than FDBSCAN and

Table 2: Clustering results in terms of ACC.

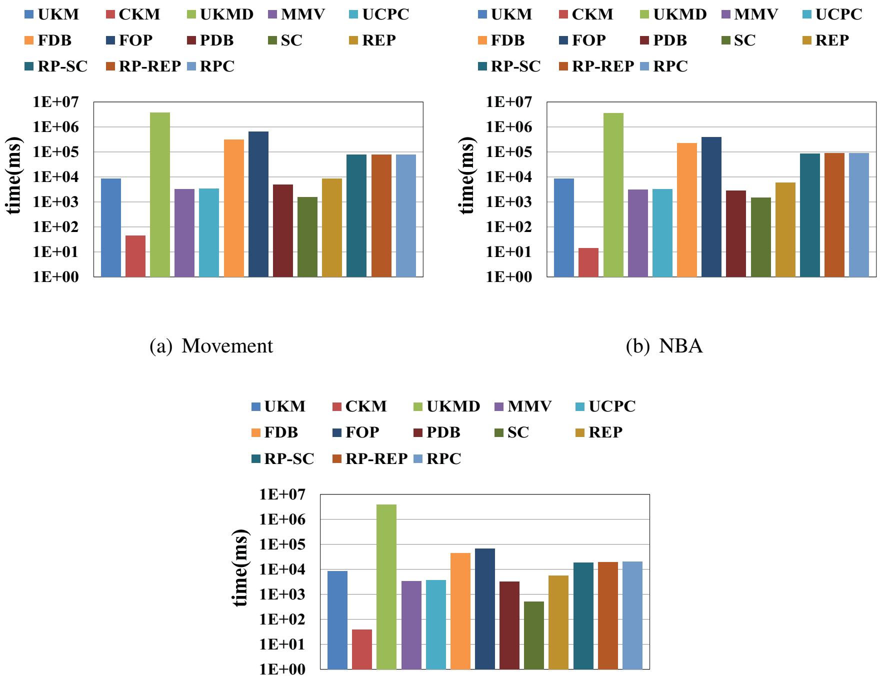
Dataset	Dist.	UKM	CKM	UKMD	MMV	UCPC	FDB	FOP	PDB	SC	REP	RP-SC	RP-REP	RPC
Wine	U	0.8180	0.8034	0.8017	0.8056	0.8354	0.7472	0.7640	0.8146	0.7079	0.7135	0.7303	0.7528	**0.9719**
	G	0.8343	0.8213	0.8163	0.8056	0.8444	0.7247	0.7528	0.7303	0.7079	0.7416	0.7360	0.8034	**0.9663**
	L	0.8567	0.8382	0.8337	0.8528	0.8500	0.7303	0.7921	0.7528	0.7022	0.7416	0.7753	0.8090	**0.9775**
Glass	U	0.4893	0.4818	0.4911	0.4313	0.4276	0.4112	0.4673	0.5047	0.4299	0.4346	0.5093	0.4953	**0.5575**
	G	0.4846	0.4897	0.4776	0.4257	0.4551	0.4533	0.4579	0.5187	0.4112	0.4299	0.4953	0.5093	**0.5565**
	L	0.4874	0.4860	0.4766	0.4322	0.4668	0.4019	0.4626	0.5000	0.4206	0.4439	0.4907	0.5000	**0.5561**
Ecoli	U	0.6661	0.6853	0.6413	0.6538	0.6557	0.5596	0.6177	0.6544	0.5810	0.6055	0.6606	0.6514	**0.8070**
	G	0.6321	0.6300	0.6352	0.6309	0.5765	0.5260	0.6667	0.6575	0.6728	0.6667	0.7034	0.7278	**0.8055**
	L	0.6489	0.5826	0.6456	0.6633	0.6419	0.5352	0.6116	0.6514	0.5902	0.6086	0.6606	0.7217	**0.8116**
Image	U	0.6872	0.6706	0.7129	0.5603	0.5621	0.5550	0.7065	0.6680	0.6108	0.6450	0.6844	0.7052	**0.8475**
	G	0.6639	0.6425	0.6980	0.5945	0.5819	0.5494	0.7177	0.7299	0.5870	0.5636	0.6576	0.6545	**0.8350**
	L	0.6724	0.6706	0.6627	0.6156	0.5808	0.5528	0.7429	0.7074	0.5264	0.5905	0.5792	0.6563	**0.8459**
Libras	U	0.5233	0.5083	0.5475	0.4231	0.4461	0.2056	0.2389	0.3139	0.2111	0.2361	0.2750	0.2944	**0.6125**
	G	0.5322	0.5053	0.5294	0.4211	0.4414	0.2528	0.3417	0.3222	0.2611	0.3167	0.3083	0.3778	**0.6006**
	L	0.5258	0.5208	0.5539	0.4236	0.4444	0.2750	0.2917	0.3306	0.2889	0.2861	0.3333	0.3472	**0.6008**
USPS	U	0.6844	0.6973	0.7097	0.5354	0.5197	0.4295	0.4790	0.5178	0.4047	0.4521	0.4930	0.5027	**0.8029**
	G	0.6220	0.6245	0.6499	0.5107	0.5269	0.4101	0.4769	0.4833	0.4101	0.4456	0.4327	0.4639	**0.7658**
	L	0.6868	0.6846	0.6226	0.5477	0.5425	0.4424	0.4822	0.4909	0.4198	0.4424	0.4822	0.5199	**0.7825**
Waveform	U	0.8335	0.8384	0.7480	0.6565	0.6626	0.3392	0.3352	0.5472	0.4274	0.4062	0.5004	0.4254	**0.9583**
	G	0.8381	0.8382	0.7080	0.6569	0.6542	0.3428	0.3294	0.5938	0.4366	0.4386	0.4956	0.4524	**0.9618**
	L	0.7797	0.7775	0.7075	0.6696	0.6843	0.3412	0.3316	0.5732	0.4248	0.4160	0.4874	0.4678	**0.9573**
Movement	—	0.3490	0.3341	0.3478	0.3427	0.3494	0.2834	0.2643	0.3121	0.2548	0.2866	0.2866	0.3153	**0.4315**
NBA	—	0.5463	0.5457	0.5403	0.5257	0.5473	0.5667	0.5067	0.5867	0.5133	0.5433	0.5667	0.5700	**0.6133**
Weather	—	0.5869	0.6144	0.5961	0.6105	0.6033	0.5882	0.5163	0.6993	0.5294	0.5490	0.6340	0.6405	**0.7176**
all avg.ACC		0.6437	0.6371	0.6314	0.5748	0.5792	0.4676	0.5147	0.5692	0.4804	0.5002	0.5407	0.5568	**0.7643**
all avg.ACC.gain		**0.1206**	**0.1272**	**0.1329**	**0.1895**	**0.1851**	**0.2967**	**0.2496**	**0.1951**	**0.2839**	**0.2641**	**0.2236**	**0.2075**	—

FOPTICS, but slower than UK-means, CK-means, MMVar, UCPC and PDBSCAN. Among possible world based algorithms, RP-SC, RP-REP and RPC perform almost

Table 3: Clustering results in terms of NMI.

Dataset	Dist.	UKM	CKM	UKMD	MMV	UCPC	FDB	FOP	PDB	SC	REP	RP-SC	RP-REP	RPC
Wine	U	0.6871	0.6002	0.6155	0.6398	0.6849	0.5303	0.6192	0.5510	0.4630	0.4762	0.5196	0.5519	**0.8926**
	G	0.7091	0.6435	0.6880	0.6419	0.6795	0.5562	0.6277	0.6195	0.4817	0.5460	0.5434	0.5823	**0.8782**
	L	0.7033	0.6649	0.7066	0.6898	0.6837	0.7002	0.7877	0.7239	0.6785	0.7384	0.7764	0.7887	**0.9088**
Glass	U	0.3335	0.3422	0.3351	0.2646	0.2850	0.3162	0.3562	0.4055	0.3693	0.3277	0.4016	0.4006	**0.4101**
	G	0.3353	0.3401	0.3377	0.2624	0.2923	0.3668	0.3627	0.4177	0.3421	0.3649	0.4025	0.4016	**0.4188**
	L	0.3580	0.3625	0.3455	0.2590	0.3074	0.3104	0.3428	0.4049	0.3536	0.3238	0.3998	0.4044	**0.4120**
Ecoli	U	0.6201	0.6055	0.6164	0.5986	0.5746	0.2335	0.5773	0.4960	0.5082	0.4357	0.5292	0.5549	**0.6824**
	G	0.6102	0.6362	0.5912	0.5569	0.5588	0.2040	0.5917	0.5536	0.4973	0.5124	0.5858	0.5860	**0.6871**
	L	0.6273	0.6384	0.5906	0.5838	0.6005	0.1898	0.5587	0.4602	0.5095	0.5121	0.5380	0.5907	**0.7074**
Image	U	0.7048	0.7041	0.6968	0.5941	0.5612	0.6661	0.7234	0.6628	0.6854	0.6607	0.7242	0.7186	**0.7756**
	G	0.7115	0.6601	0.6818	0.6070	0.5933	0.6849	0.7464	0.7647	0.6182	0.5871	0.6925	0.6854	**0.7838**
	L	0.7160	0.7016	0.6914	0.6386	0.6465	0.6836	0.7761	0.7403	0.6001	0.6235	0.6707	0.7026	**0.7970**
Libras	U	0.6595	0.6329	0.6352	0.5431	0.5703	0.4249	0.4251	0.6161	0.4096	0.3999	0.4642	0.5298	**0.6841**
	G	0.6583	0.6555	0.6314	0.5490	0.5752	0.4814	0.5742	0.5637	0.4997	0.5752	0.5292	0.6100	**0.7056**
	L	0.6580	0.6535	0.6530	0.5590	0.5739	0.5024	0.5298	0.5662	0.5049	0.4693	0.5327	0.5626	**0.7019**
USPS	U	0.7661	0.7593	0.7438	0.5544	0.5651	0.5103	0.6064	0.6842	0.4251	0.5057	0.5658	0.5587	**0.8473**
	G	0.6797	0.6539	0.6574	0.5338	0.5326	0.4741	0.5622	0.5782	0.4939	0.5386	0.5242	0.5639	**0.8082**
	L	0.7507	0.7413	0.6529	0.5433	0.5414	0.5221	0.5236	0.6035	0.5032	0.5651	0.5388	0.5936	**0.8120**
Waveform	U	0.6572	0.6662	0.5333	0.3632	0.4034	0.0676	0.0755	0.1949	0.1001	0.1005	0.1138	0.1107	**0.8281**
	G	0.6667	0.6697	0.4554	0.4397	0.4046	0.0602	0.0667	0.2975	0.0931	0.0489	0.1061	0.1465	**0.8400**
	L	0.5964	0.5886	0.4557	0.4090	0.4477	0.0506	0.0858	0.2245	0.0972	0.1385	0.1088	0.1579	**0.8259**
Movement	—	0.2133	0.1935	0.2172	0.1837	0.1985	0.0445	0.0791	0.1170	0.0688	0.0975	0.1350	0.1361	**0.2584**
NBA	—	0.1591	0.1648	0.1558	0.1647	0.1690	0.1443	0.0918	0.1759	0.0671	0.1446	0.1563	0.1571	**0.1919**
Weather	—	0.4892	0.4690	0.4486	0.4183	0.3747	0.1937	0.4575	0.5277	0.2714	0.2761	0.3133	0.3724	**0.5842**
all avg.NMI		0.5863	0.5728	0.5473	0.4832	0.4927	0.3716	0.4645	0.4979	0.4017	0.4154	0.4530	0.4778	**0.6851**
all avg.NMI.gain		**0.0988**	**0.1123**	**0.1378**	**0.2019**	**0.1924**	**0.3135**	**0.2206**	**0.1872**	**0.2834**	**0.2697**	**0.2321**	**0.2073**	—

identically, and they are slower than SC and REP. The reason is that when selecting representative possible worlds, the computation process of Jensen-Shannon divergence



(a) Movement

(b) NBA

(c) Weather

Figure 4: Clustering results in terms of efficiency.

is a little complex. In summary, RPC performs acceptably in efficiency.

6. Conclusion

In this paper, we propose a representative possible world based consistent clustering algorithm for uncertain data. It consists of two parts: selecting representative possible worlds and consistent spectral clustering. By selecting representative possible worlds, it avoids the negative effects caused by marginal possible worlds. By consistent spectral clustering, it makes use of the consistency principle to achieve better performance. Experimental results show that the proposed algorithm outperforms the state-of-the-art algorithms in effectiveness and performs competitively in terms of efficiency. For future work, we will extend the idea to uncertain data stream clustering and classification, and apply our method in more real applications.

Acknowledgment

The authors are grateful to the editor in chief, the associate editor and the reviewers for their valuable comments and suggestions. This work was supported by National Natural Science Foundation of China (No. 61876028), the grants 1-ZVJJ and G-YBXV funded by the Hong Kong Polytechnic University, and the Fundamental Research Funds for the Central Universities (No. DUT20RC(3)040).

References

[1] C. D. Manning, P. Raghavan, H. Schütze, Introduction to Information Retrieval, Cambridge University Press, 2008.

[2] C. C. Aggarwal, P. S. Yu, A survey of uncertain data algorithms and applications, IEEE Transactions on Knowledge and Data Engineering 21 (5) (2009) 609–623.

[3] X. Zhang, H. Liu, X. Zhang, Novel density-based and hierarchical density-based clustering algorithms for uncertain data, Neural Networks 93 (2017) 240–255.

[4] C. Guo, R. Zhuang, Y. Jie, K. R. Choo, X. Tang, Secure range search over encrypted uncertain iot outsourced data, IEEE Internet Things Journal 6 (2) (2019) 1520–1529.

[5] C. Lai, T. Wang, C. Liu, L. Wang, Probabilistic top-k dominating query monitoring over multiple uncertain iot data streams in edge computing environments, IEEE Internet Things Journal 6 (5) (2019) 8563–8576.

[6] H. Liu, X. Zhang, X. Zhang, Possible world based consistency learning model for clustering and classifying uncertain data, Neural Networks 102 (2018) 48–66.

[7] H. Liu, X. Zhang, X. Zhang, Pwadaboost: Possible world based adaboost algorithm for classifying uncertain data, Knowledge Based Systems. 186.

[8] K. K. Sharma, A. Seal, Modeling uncertain data using monte carlo integration method for clustering, Expert Systems with Applications 137 (2019) 100–116.

[9] J. Hou, Y. Li, J. Yu, W. Shi, A survey on digital forensics in internet of things, IEEE Internet Things Journal 7 (1) (2020) 1–15.

[10] M. Chau, R. Cheng, B. Kao, J. Ng, Uncertain data mining: An example in clustering location data, in: Proceedings of PAKDD, 2006, pp. 199–204.

[11] F. Gullo, G. Ponti, A. Tagarelli, Clustering uncertain data via K-medoids, in: Proceedings of SUM, 2008, pp. 229–242.

[12] H.-P. Kriegel, M. Pfeifle, Density-based clustering of uncertain data, in: Proceedings of KDD, 2005, pp. 672–677.

[13] H.-P. Kriegel, M. Pfeifle, Hierarchical density-based clustering of uncertain data, in: Proceedings of ICDM, 2005, pp. 689–692.

[14] M. Ester, H.-P. Kriegel, J. Sander, X. Xu, A density-based algorithm for discovering clusters in large spatial databases with noise, in: Proceedings of KDD, 1996, pp. 226–231.

[15] M. Ankerst, M. M. Breunig, H.-P. Kriegel, J. Sander, OPTICS: Ordering points to identify the clustering structure, in: Proceedings of SIGMOD, 1999, pp. 49–60.

[16] A. Züfle, T. Emrich, K. A. Schmid, N. Mamoulis, A. Zimek, M. Renz, Representative clustering of uncertain data, in: Proceedings of KDD, 2014, pp. 243–252.

[17] P. B. Volk, F. Rosenthal, M. Hahmann, D. Habich, W. Lehner, Clustering uncertain data with possible worlds, in: Proceedings of ICDE, 2009, pp. 1625–1632.

[18] H. Liu, X. Zhang, X. Zhang, Q. Li, X. Wu, Clustering uncertain data via representative possible worlds with consistency learning, in: Proceedings of IJCAI Workshops, 2019.

[19] B. Kao, S. D. Lee, D. W. Cheung, W.-S. Ho, K. F. Chan, Clustering uncertain data using Voronoi diagrams, in: Proceedings of ICDM, 2008, pp. 333–342.

[20] B. Kao, S. D. Lee, F. K. F. Lee, D. W.-L. Cheung, W.-S. Ho, Clustering uncertain data using Voronoi diagrams and R-tree index, IEEE Transactions on Knowledge and Data Engineering 22 (9) (2010) 1219–1233.

[21] W. K. Ngai, B. Kao, R. Cheng, M. Chau, S. D. Lee, D. W. Cheung, K. Y. Yip, Metric and trigonometric pruning for clustering of uncertain data in 2D geometric space, Information Systems 36 (2) (2011) 476–497.

[22] I. Lukic, M. Köhler, N. Slavek, Improved bisector pruning for uncertain data mining, in: Proceedings of ITI, 2012, pp. 355–360.

[23] S. D. Lee, B. Kao, R. Cheng, Reducing UK-means to K-means, in: Proceedings of ICDM Workshops, 2007, pp. 483–488.

[24] J. Zhou, L. Chen, C. L. P. Chen, Y. Wang, H. Li, Uncertain data clustering in distributed peer-to-peer networks, IEEE Transactions on Neural Networks and Learning Systems 29 (6) (2018) 2392–2406.

[25] F. Gullo, G. Ponti, A. Tagarelli, Minimizing the variance of cluster mixture models for clustering uncertain objects, in: Proceedings of ICDM, 2010, pp. 839–844.

[26] F. Gullo, A. Tagarelli, Uncertain centroid based partitional clustering of uncertain data, in: Proceedings of VLDB, 2012, pp. 610–621.

[27] E. Schubert, A. Koos, T. Emrich, A. Züfle, K. A. Schmid, A. Zimek, A framework for clustering uncertain data, in: Proceedings of VLDB, 2015, pp. 1976–1979.

[28] X. Zhang, X. Zhang, H. Liu, X. Liu, Multi-task multi-view clustering, IEEE Transactions on Knowledge and Data Engineering (TKDE) 28 (12) (2016) 3324–3338.

[29] S. Bickel, T. Scheffer, Multi-view clustering., in: Proceedings of ICDM, 2004, pp. 19–26.

[30] X. Liu, M. Li, L. Wang, Y. Dou, J. Yin, E. Zhu, Multiple kernel k-means with incomplete kernels, in: Proceedings of AAAI, 2017, pp. 2259–2265.

[31] P. Yang, H. Davulcu, Y. Zhu, J. He, A generalized hierarchical multi-latent space model for heterogeneous learning, IEEE Transactions on Knowledge and Data Engineering 28 (12) (2016) 3154–3168.

[32] W. Wang, Z. Zhou, A new analysis of co-training, in: Proceedings of ICML, 2010, pp. 1135–1142.

[33] S. Dasgupta, M. L. Littman, D. A. McAllester, PAC generalization bounds for co-training, in: Proceedings of NIPS, 2001, pp. 375–382.

[34] A. D. Sarma, O. Benjelloun, A. Y. Halevy, S. U. Nabar, J. Widom, Representing uncertain data: Models, properties, and algorithms, VLDB Journal 18 (5) (2009) 989–1019.

[35] N. N. Dalvi, D. Suciu, Management of probabilistic data: Foundations and challenges, in: Proceedings of PODS, 2007, pp. 1–12.

[36] M. Hua, J. Pei, Ranking Queries on Uncertain Data, Advances in Database Systems, Springer Press, 2011.

[37] L. Devroye, Non-uniform Random Variate Generation, Springer Press, 1986.

[38] R. Jampani, F. Xu, M. Wu, L. L. Perez, C. Jermaine, P. J. Haas, MCDB: A Monte Carlo approach to managing uncertain data, in: Proceedings of SIGMOD, 2008, pp. 687–700.

[39] J. Lin, Divergence measures based on the Shannon entropy, IEEE Transactions on Information Theory 37 (1) (1991) 145–151.

[40] S. Kullback, R. A. Leibler, On information and sufficiency, The Annals of Mathematical Statistics 22 (1) (1951) 79–86.

[41] B. W. Silverman, Density Estimation for Statistics and Data Analysis, CRC Press, 1986.

[42] B. Baker, H. Diaz, W. Hargrove, F. Hoffman, Use of the Köppen–Trewartha climate classification to evaluate climatic refugia in statistically derived ecoregions for the People's Republic of China, Climatic Change 98 (1–2) (2010) 113–131.