Representative Clustering of Uncertain Data

Andreas Züfle*   Tobias Emrich*   Klaus Arthur Schmid*   Nikos Mamoulis+
Arthur Zimek*   Matthias Renz*
*Institute for Informatics, Ludwig-Maximilians-Universität München
{zuefle,emrich,schmid,zimek,renz}@dbs.ifi.lmu.de
+Department of Computer Science, University of Hong Kong
nikos@cs.hku.hk

ABSTRACT

This paper targets the problem of computing meaningful clusterings from uncertain data sets. Existing methods for clustering uncertain data compute a single clustering without any indication of its quality and reliability; thus, decisions based on their results are questionable. In this paper, we describe a framework, based on possible-worlds semantics; when applied on an uncertain dataset, it computes a set of representative clusterings, each of which has a probabilistic guarantee not to exceed some maximum distance to the ground truth clustering, i.e., the clustering of the actual (but unknown) data. Our framework can be combined with any existing clustering algorithm and it is the first to provide quality guarantees about its result. In addition, our experimental evaluation shows that our representative clusterings have a much smaller deviation from the ground truth clustering than existing approaches, thus reducing the effect of uncertainty.

1. INTRODUCTION

In a variety of application domains, our ability to unearth a wealth of new knowledge from a data set is impaired by unreliable, erroneous, obsolete, imprecise, and noisy data. Reasons and sources of such uncertainty are many. Sensing devices are inherently imprecise (e.g., due to signal noise, instrumental errors and transmission errors [15]). Moving objects can only be monitored sporadically, such that at a certain time the position of an object is not explicitly known [50]. Integration of data from heterogeneous sources may incur uncertainty, for example due to uncertain schema matchings between different data sources [3]. Uncertainty may also be injected to the data on purpose, for privacy preservation reasons [19].

Ignoring data uncertainty in a mining task (e.g., replacing any uncertain values by their expectations) may compromise the quality of the result. On the other hand, by considering the uncertainty directly in the mining process, we can assess the reliability of the result, giving the user a notion about its quality and giving an intuition of how likely it is identical, or at least similar, to the result of the mining task when applied to the true (but unknown) data values [48]. For instance, in association rule mining on uncertain data, confidence values of the probability that a given itemset is frequent are derived [5]. This notion of confidence allows the user to make a more educated judgement of the data, thus enhancing the underlying decision-making process.

This paper targets the problem of deriving a meaningful clustering from an uncertain dataset. For this purpose, our aim is not to develop a new clustering algorithm, but rather to allow clustering algorithms designed for certain data to return meaningful, reliable and correct results in the presence of uncertainty. To illustrate the challenge that arises by considering uncertain data, consider the work-flow depicted in Figure 1. Figure 1(a) shows a dataset containing six two-dimensional points, which correspond to the positions of moving objects at some point of time t. The shaded region in Figure 1(a) corresponds to a lake, which none of the moving objects may cross. To cluster the locations of these objects, a domain expert may opt to choose a clustering algorithm $\mathcal{C}$ from a suite of available options (e.g., a density-based algorithm [34], such as DBSCAN [17] or HDBSCAN [8], or one of the numerous variants from the k-means family [27]). Assuming that the true locations of the objects are known, $\mathcal{C}$ can compute the clustering shown in Figure 1(b). However, the true locations of the objects could be unknown and we may only have access to the last reported observations of these objects (e.g., by their GPS devices), shown as triangles in Figure 1(c). In such a scenario, an *uncertainty data model* is typically used to capture the distribution of the possible object locations. For instance, past observations as well as empirically learned moving patterns of an object can be used to obtain a probability function for the position at a time after the object's last observation [16]. Examples of *probability density functions* (PDFs) around the observations are shown in the Figure.

Object A, for instance, is likely to be moving around the lake (since movement inside the lake is impossible), while the movements of other objects are less constrained. If we follow a simplistic approach for clustering the data, by clustering the expected values of the objects according to the uncertain data model, then we may end up in deriving a clustering as shown in Figure 1(c) which arbitrarily deviates from the clustering of the actual (but unknown) locations of the objects shown in Figure 1(b). Therefore, approaches that aggregate the information given by the uncertain model to expected values and then apply clustering may yield results
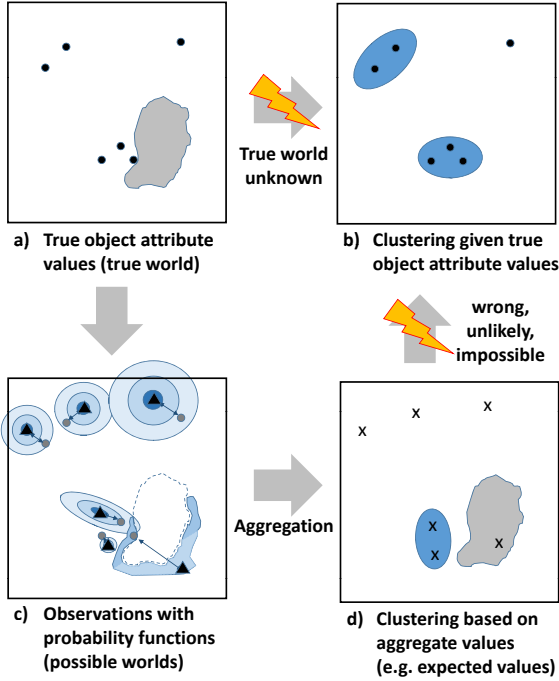
Figure 1: Uncertain Clustering Workflow.

of poor quality, due to information loss and potentially due to invalid input to the clustering process (e.g., the expected location of A after aggregating all its possible positions in Figure 1(d) is in the lake).

In this paper, our goal is neither the definition of new uncertain data models suited for clustering, nor the proposal of new clustering algorithms tailored for uncertain data. Instead, we aim at making possible the application of *any* existing clustering algorithm $\mathcal{C}$ on an uncertain database $\mathcal{DB}$, based on *any* given uncertain data model for $\mathcal{DB}$, assuming that $\mathcal{C}$ would be appropriate for $\mathcal{DB}$, if $\mathcal{DB}$ was certain. For example, we take that the clustering of Figure 1(b) by algorithm $\mathcal{C}$ is the ideal one, but infeasible to derive, given that we do not know the actual locations of the objects. The objective of our framework is to use the data of Figure 1(c) and algorithm $\mathcal{C}$ to derive a clustering that has a probabilistic guarantee (according to the uncertain data model used) to be very similar to that of Figure 1(b).

Our approach performs sampling and then represents the original uncertain database $\mathcal{DB}$ as a set of sample deterministic databases. We then run the clustering algorithm $\mathcal{C}$ on each of the sampled databases, to derive a set of possible clusterings PC. Our main contribution is to combine the resulting set PC into a concise set RPC of *representative clusterings*. Furthermore, using all clusterings in PC, we estimate a probability ϕ of each representative clustering in RPC, defined as the probability that this representative clustering does not exceed some maximum distance τ to the true clustering. Since ϕ has to be estimated from sampled databases, we obtain a lower bound of ϕ that is significant at a user specified level. We provide a methodology to derive an RPC for a given value of τ; and therefore impose quality constraints on the uncertain clustering results, unlike previous approaches on uncertain data clustering [41, 21, 23, 38, 32, 22, 31, 36, 37] which cannot provide quality guarantees.

In summary, our contributions are as follows.

- We propose a sampling-based solution to cluster uncertain data. This solution is generally applicable to all data domains and with any suitable uncertain data model, allowing the application of any existing clustering algorithm, originally designed for certain data. As opposed to previous work on uncertain clustering, our approach conforms to the possible worlds semantics and also considers any dependencies between objects.
- We present a methodology via which we can assess the quality of a clustering of a possible world compared to the true clustering of the data.
- We show how the confidence of clustering results on possible worlds can be improved by computing a set of multiple representative clusterings, each having a significant likelihood to resemble the true, unknown clustering.

The rest of the paper is organized as follows. Section 2 surveys existing methods for clustering uncertain data. Section 3 gives general definitions used in the remainder of this work. Section 4 shows how we can estimate the probability of the clustering result on a possible world to be the clustering of the true data values. Section 5 shows how, from a set of possible clusterings, we can find *representative clusterings* that are probabilistically guaranteed to be similar to the real clustering. Section 6 evaluates our framework experimentally. Section 7 concludes this work.

2. RELATED WORK

Clustering is undoubtedly one of the most important tools for unsupervised classification. A large number of clustering algorithms has been developed, as reflected in numerous surveys [28, 34, 45]. Although clustering has proved its applicability in many different domains and scenarios, the problem of clustering uncertain data has only gained little attention so far. Uncertain data clustering approaches either use expected distances between objects or assume that the distances between different pairs of objects are independent. In this work we review these methods and discuss their drawbacks.

Clustering Using Expected Distances. The main drawback of approaches based on expected distances [41, 21, 23] is the information loss incurred by describing a complex probability distance function by a single scalar. Considering additional moments of a probability distance function, such as deviation [23] works well in specific applications where objects have little uncertainty. Still, the quality of such approaches cannot be assessed, rendering them inappropriate for applications where decisions are to be made based on the computed clustering. As an example consider the setting of Figure 2, having two certain objects A and B, and an uncertain object U having two possible values U_1 and U_2. Now, assume a deterministic clustering algorithm $\mathcal{C}$ which clusters two objects only if there distance does not exceed $dist(A, U_1)$ ($= dist(B, U_2)$). Clearly, in the example of Figure 2, there are two possible clusterings, either the clustering having cluster $\{A, U\}$ and outlier B, or the clustering having cluster $\{U, B\}$ and outlier A. The probabilities of these possible clusterings equal the probabilities $0 < P(U_1) < 1$ and $P(U_2) = 1 - P(U_1)$ of alternatives U_1 and U_2 of U. However, using the expected distance between A and U given by

$$E(dist(A, U)) = P(U_1) \cdot dist(A, U_1) + P(U_2) \cdot dist(A, U_2),$$

Figure 2: Example of Uncertainty

objects A and U can never be located in the same cluster, since it holds that $E(dist(A, U)) > dist(A, U_1)$. The same holds for B and U. Thus, for the example of Figure 2, an approach using expected distance yields a clustering, which is strictly impossible. Summarizing, the use of expected distances is a heuristic to obtain a single clustering that represents the whole realm of possible clusterings, however, the derived result is not necessarily similar to any of them. Most clustering algorithms using expected distances focus on improving efficiency rather than addressing this drawback [38, 32, 41]. Gullo et al. [22] propose a method, called UK-medoids, for which they utilize the expected distance between two uncertain objects. Jiang et al. [31] propose to use the KL divergence between two uncertain objects which is often used to reflect similarity between two probabilistic density functions. They investigate the applicability to both K-means [40] and DBSCAN [17]. Since all these approaches are based on expected distances, their results are not in accordance with the possible worlds semantics and do not carry any quality guarantee.

Clustering Assuming Independent Distances. Kriegel and Pfeifle [36, 37] assume that pairwise distances between uncertain objects are mutually independent. This assumption may yield wrong results; in addition, the results are biased toward overestimating the size of clusters. In particular, they rely on the following assumption on distances between uncertain objects:

$$P(A \leftrightarrow_\epsilon B \wedge B \leftrightarrow_\epsilon C) = P(A \leftrightarrow_\epsilon B) \cdot P(B \leftrightarrow_\epsilon C).$$

$P(X \leftrightarrow_\epsilon Y)$ denotes the probability that the distance between the two uncertain objects X and Y is smaller than a threshold ϵ. However this assumption does not hold in the general case. For example, in Figure 2, the two random events $A \leftrightarrow_\epsilon U$ and $U \leftrightarrow_\epsilon B$ are negatively correlated: for the case where $dist(A, U_1) < \epsilon < dist(A, U_2)$, it even holds that both random events are mutually exclusive. That is, if U is close to A then it cannot be close to B and vice versa. Relaxing this assumption of independent distances yields a computationally hard problem, as a distance value may depend on a large number of uncertain objects.

Discussion. To our knowledge, there is no previous work on uncertain data clustering that conforms to the possible worlds semantics. A likely reason for this that general induction on uncertain data is a #P-hard problem [13]. In order to avoid the exponential run-time cost of considering all possible worlds, a common approach to handle uncertain data, in general, is sampling [30]. Given a sample of instances of the database (each corresponding to a possible world), a query or data mining task can be performed on each of them, and common results can be returned, associated with confidences of these results to be equal (or sufficiently similar) to the result on the true (but unknown) data. This is exactly the approach that we are following in this paper. We provide a small number of representative clusterings that have high confidence values to be similar to the true (but unknown) clustering.

3. DEFINITIONS

This section gives definitions for uncertain databases and clustering, which generalize all previous models and definitions. While existing works [37, 36, 38, 9, 23] generally assume that objects are points in a multi-dimensional feature space, we allow objects to have any abstract type $\mathcal{O}$, where $\mathcal{O}$ is an object space, such as a multi-dimensional feature space, the set of all strings, the set of all images, etc. Each *uncertain object* is then represented by a set of values in $\mathcal{O}$, each associated with a non-zero probability. This probability distribution can be a continuous *probability density function* (PDF) or a discrete *probability mass function* (PMF). To model the uncertainty of an object in a general way that captures both continuous [36, 42] and discrete [49, 42] models, we use the following general definition.

DEFINITION 1 (MULTIVARIATE UNCERTAIN OBJECT).
A multivariate uncertain object o is defined by two functions $\text{pdf}_o : \mathcal{O} \to \mathbb{R}_0^+$ *and* $\text{pmf}_o : \mathcal{O} \cup \perp \to [0, 1]$ *such that*

$$\int_\mathcal{O} \text{pdf}_o(x)dx + \sum_{\mathcal{O} \cup \perp} \text{pmf}_o(x) = 1$$

Value $\perp$ is used to model existentially uncertainty, i.e., with a probability of $\text{pmf}_o(\perp)$ object o does not exist at all in the database. By setting either $\text{pdf}_o(x)$ or $\text{pmf}_o(x)$ to the zero function, which maps any value to zero, the above definition can simulate discrete and continuous models, while also allowing mixed models.[1]

Consequently, if there is at least one uncertain object in a database, the state of the database becomes a random variable. To model the semantics of a database being a random variable, the concept of *possible worlds* is commonly used [43, 46, 54, 39]: an uncertain database is defined by a (potentially infinite) set of possible database states, called *possible worlds*. Each possible world is associated with its corresponding probability to be the true database state.

DEFINITION 2 (UNCERTAIN DATABASE).
An uncertain database $\mathcal{DB}$ is defined by a set of uncertain objects $\mathcal{DB} = \{o_1, ..., o_{|\mathcal{DB}|}\}$ spanning a (potentially infinite) set of possible worlds $\mathcal{W}$ and a constructive generation rule $\mathcal{G}$ to draw possible worlds from $\mathcal{W}$ in an unbiased way. The probability to draw a world w equals the probability $P(w)$ of this world being the true (but unknown) world oracle$(\mathcal{DB})$.

Wherever independence is assumed between objects [25, 6, 10], the generation rule $\mathcal{G}$ is implicitly given by drawing samples from each object individually. In scenarios with interdependencies between uncertain objects (for example, expressed by a Bayesian network [44], or by lineage of relations [2]), a possible world can be drawn by using the factorized representation of the Bayesian network, and iteratively drawing objects conditioned to previously drawn objects.[2] To the best of our knowledge, our assumption of having a generation rule $\mathcal{G}$ is met in all state-of-the-art uncertain database management systems. The task of clustering can be defined as follows.

---

[1] Note that this definition avoids the (wrong) claim that a discrete distribution can be seen as a special case of a continuous distribution without any form of continuity correction.
[2] The factorized representation guarantees that at each iteration there must be one random variable for which the required conditions are met.

DEFINITION 3 (CLUSTERING).
A clustering $\mathcal{C}(S)$ of a set $S = \{a_1, ..., a_N\}$ of deterministic objects is a partitioning of S into pairwise disjoint subsets $C_1, ..., C_k \subseteq S$, such that $\bigcup_{1 \leq i \leq k} C_i = S$. Each subset $C_i, 1 \leq i \leq k$ is called a cluster.

This abstract definition of clustering intentionally omits any objective rules toward a "good clustering", such as the requirement that similar objects should be in the same cluster. The reason is that our approach should be used in conjunction with any clustering algorithm $\mathcal{C}$, independently to the algorithm's objective. Due to the theoretical result that general query processing (or mining) on uncertain data is $\#P$-complete [13], coupled with the fact that an uncertain database may yield a number of possible clusterings exponential in the size of the database, we now explore the possibility of using a Monte-Carlo approach to perform clustering over uncertain data.

4. CLUSTERING SAMPLED WORLDS

Let $\mathcal{DB}$ be an uncertain database and let $\mathcal{C}$ be a clustering algorithm. Let $X = \{X_1, ..., X_{|X|}\}$ be a multiset[3] of possible worlds of $\mathcal{DB}$ generated from $\mathcal{DB}$ using generation rule $\mathcal{G}$ and let $\mathcal{C}(X)$ denote the multiset of clusterings obtained by clustering each sample world in X. We denote the set of distinct clusterings in $\mathcal{C}(X)$ as the set PC of *possible clusterings* obtained from sample X. For any clustering C in PC, the *support $C.supp$* of C is defined as $\sum_{i=1}^{|X|} I(\mathcal{C}(X_i) = C)$, where $I(\cdot)$ is an indicator function that returns 1 if its operand is true and 0, otherwise. Simply speaking, $C.supp$ is the number of occurrences of clustering C in the multiset $\mathcal{C}(X)$.

LEMMA 1 (APPROXIMATION OF THE TRUE CLUSTERING).
For any possible clustering $C \in PC$, the probability $\hat{P}(C) = \frac{C.supp}{|X|}$ is an unbiased estimator the probability $P(C)$ that C is the true clustering $\mathcal{C}(\mathrm{oracle}(\mathcal{DB}))$ of $\mathcal{DB}$; i.e., $E(\hat{P}(C)) = P(C)$.

PROOF.

$$E(\hat{P}(C)) = E\left(\frac{C.supp}{|X|}\right) = E\left(\frac{\sum_{X_i \in X} I(X_i = C)}{|X|}\right),$$

Since the expectation of a non-random variable is the identity, we obtain

$$E(\hat{P}(C)) = \frac{E\left(\sum_{X_i \in X} I(X_i = C)\right)}{|X|},$$

Since all sample databases X_i are drawn independently, and since the expectation of a sum of independent random variables is the sum of their expectations, we get:

$$E(\hat{P}(C)) = \frac{\sum_{X_i \in X} E(I(X_i = C))}{|X|}$$

Due to the assumption that each sample X_i is drawn unbiased from the distribution of all worlds of $\mathcal{DB}$, which implies that $E(I(X_i = C)) = P(X_i)$, we obtain

$$\frac{\sum_{X_i \in X} P(X_i)}{|X|} = P(X_i)$$

□

---

[3]Due to independent sampling, the same sample may be drawn multiple times.

Such a straightforward sampling approach works well for small databases, including the running example depicted in Figure 2, where the number of possible clusterings C is reasonably small. In a large database setting, where the probability of finding exactly the same clustering on two samples in X approaches zero, this approach becomes inapplicable. The reason is twofold. First, the probabilities $P(C)$ of a clustering C being the true clustering of $\mathcal{DB}$, become very small. Due to independent samplings $X_i, 1 \leq i \leq |X|$, the number of samples where $\mathcal{C}(X_i) = C$ follows a binomial $B(\pi = P(C), n = |X|)$ distribution. Estimating the probability parameter π of a binomial distribution given a sample, requires a very large sample size n if π is small. A rule of thumb is that $n \cdot \pi \geq 5$ [11, 52]. Second, the large number of possible clusterings combined with small probabilities makes the exact results meaningless for a user. A huge set of possible clusterings, potentially exponentially large in the number of uncertain objects, where many may be very similar, yet different between each other, is of little use.

5. REPRESENTATIVE CLUSTERINGS

Our goal is to reduce the (potentially huge) set of clusterings produced by the Monte-Carlo approach to a small set of possible clusterings, which are diverse and at the same time guaranteed to be similar to the clustering on the real (but unknown) database. In Section 5.1, we discuss a general concept for determining one representative from a set, the medoid approach [47]. In Section 5.2, we generalize this approach to select a set of multiple representative clusterings and show how we can estimate how well they can approximate the real clustering.

It is a common trend in the clustering community to provide several, different ("alternative" [20, 18, 14]) results rather than just one. On the other hand, it is also consensus to avoid an abundance of redundant results [35, 55]. The eminent question is then, how many solutions to provide and how representative these solutions are. For the problem of clustering uncertain data, we therefore present, in Section 5.3, a methodology for selecting a set of representative clusterings of guaranteed quality.

5.1 Sample Medoid

Let PC denote the set of possible clusterings derived from sampled worlds $X = \{X_1, ..., X_n\}$. Let D be the *distance* $|PC| \times |PC|$ matrix such that

$$D_{i,j} := \mathrm{dist}(PC_i, PC_j).$$

Here, dist denotes a distance measure between two clusterings, such as, e.g., the Adjusted Rand Index (ARI) [26]. Similarity usually takes a value between 0 (no agreement) and 1 (identical partitionings) and can be converted to a distance after subtraction from 1.

The *median* of PC can be defined as

$$\mathrm{Median}(PC) = \mathrm{Median}_{X_i \in X}(\mathcal{C}(X_i)) =$$

$$\arg\min_i \sum_{j=1}^n \mathrm{dist}(\mathcal{C}(X_i), \mathcal{C}(X_j)) =$$

$$\arg\min_i \sum_{j=1}^n D_{i,j} \cdot PC_i.supp$$

Arguably, the median clustering can be the most representative clustering out of all sampled clusterings $\mathcal{C}(X_i)$. How-

ever, we do not have confidence information for Median(PC), i.e., the deviation of the true clustering $\mathcal{C}(\text{oracle}(\mathcal{DB}))$ from Median(PC) is impossible to assess.

It is important to note that Median(PC), albeit derived using expected distances between clusterings, does not suffer from the same drawbacks as existing works on clustering uncertain data using expected object positions and expected distances (cf. Section 2). The main difference is that Median(PC) is a clustering derived from a possible database instance that was generated consistently to the uncertainty data model, i.e., considering the value distributions and stochastic dependencies between objects.

5.2 Multiple Representatives

The possible clusterings of an uncertain database may be very heterogeneous; depending on object attribute values of a world, an individual cluster may become noise, may shatter into multiple clusters, or may be absorbed by another cluster in some worlds, but not in others. Such large changes in the overall clustering may be caused by minimal changes in the underlying dataset: the density of a critical region may drop below the threshold of a density-based clustering algorithm; a partition-based cluster representative may change slightly, yielding a new data partitioning and leading into a spiral of changes. Keeping this potential heterogeneity of possible clusterings in mind, a single sample medoid clustering could be insufficient: it may be an unlikely pivot between a number of likely clusterings and it may not even be similar to the most likely possible worlds. Instead, a user may be more interested in a smaller set of clusterings, all having a significantly high probability to be similar (but not necessarily equal) to the true clustering, i.e., being representative. We define a representative clustering as follows:

DEFINITION 4   (REPRESENTATIVE CLUSTERING).
Let $\mathcal{DB}$ be an uncertain database and let $\mathcal{C}$ be a clustering algorithm. We call a clustering $\mathcal{C}(X_i)$ a τ-ϕ-representative clustering, if the probability

$$P(X_i, \tau) := P(\text{dist}(\mathcal{C}(X_i), \mathcal{C}(\text{oracle}(\mathcal{DB}))) \leq \tau)$$

that the true clustering $\mathcal{C}(\text{oracle}(\mathcal{DB}))$ of $\mathcal{DB}$ has a distance $\text{dist}(\mathcal{C}(X_i), \mathcal{C}(\text{oracle}(\mathcal{DB})))$ of at most τ is at least ϕ.

LEMMA 2   (APPROXIMATION OF REPRESENTATIVES).
Let $X = \{X_1, ..., X_{|X|}\}$ be a set of possible worlds of $\mathcal{DB}$ generated from $\mathcal{DB}$ using generation rule $\mathcal{G}$ and let dist be a distance measure on clusterings. Let PC be the set of clusterings obtained from X associated with their supports. The probability

$$\hat{P}(X_i, \tau) := \frac{\sum_{j=1}^{|X|} I(\text{dist}(\mathcal{C}(X_i), \mathcal{C}(X_j)) \leq \tau)}{|X|}$$

is an unbiased estimator of the probability

$$P(X_i, \tau) := P(\text{dist}(\mathcal{C}(X_i), \mathcal{C}(\text{oracle}(\mathcal{DB}))) \leq \tau)$$

that cluster representative X_i has a distance of at most τ to the true clustering of $\mathcal{DB}$.

PROOF. Analogous to Lemma 1, by substituting the predicate $(\text{dist}(\mathcal{C}(X_i), \mathcal{C}(X_j)) \leq \tau)$ for $(C = X_i)$. $\square$

Albeit unbiased, the probability $\hat{P}(X_i, \tau)$ cannot be used directly to assess the probability $P(X_i, \tau)$ of cluster X_i having

a distance of at most τ to the true clustering $\mathcal{C}(\text{oracle}(\mathcal{DB}))$. Thus, X_i can not simply be returned as a τ-$\phi = \hat{P}(X_i, \tau)$-representative according to Definition 4, because the estimator $\hat{P}(X_i, \tau)$ may overestimate the true probability $P(X_i, \tau)$. To return τ-ϕ representative clusters to the user, our aim is to find a lower bound $\hat{P}(X_i, \tau, \alpha)$ such that we can guarantee that $P(X_i, \tau) \geq \hat{P}(X_i, \tau, \alpha)$ with a probability of α, where α is a domain specific level of significance (typically, $\alpha = 0.95$).

To derive such a significant lower bound of $P(X_i, \tau)$ we may exploit the fact that sampled possible worlds were drawn independently. Therefore, the absolute number $\hat{P} \cdot |X|$ of sampled worlds which are represented by X_i follows a binomial $B(P(X_i, \tau), |X|)$ distribution. To estimate the true probability $P(X_i, \tau)$, given realization $\hat{P} \cdot |X|$, we borrow techniques from statistics to obtain a one sided $1 - \alpha$ confidence interval of the true probability $P(X_i, \tau)$. A simple way of obtaining such confidence interval is by applying the central limit theorem to approximate a binomial distribution by a normal distribution.

DEFINITION 5   (α-CONFIDENCE PROBABILITIES).
Let $\mathcal{DB}$ be an uncertain database. For a set of drawn database instances X, and for a possible clustering $\mathcal{C}(X_i), X_i \in X$, a distance threshold τ and a level of significance α, the probability

$$\hat{P}(X_i, \tau, \alpha) = \hat{P}(X_i, \tau) - z \cdot \sqrt{\frac{1}{|X|} \hat{P}(X_i, \tau) \left(1 - \hat{P}(X_i, \tau)\right)},$$

is called α-confidence probability of τ-representative X_i, where z is the $100 \cdot (1 - \alpha)$ percentile of the standard normal distribution.

The α-confidence probability $\hat{P}(X_i, \tau, \alpha)$ can be used to return the clustering $\mathcal{C}(X_i)$ as a τ-ϕ-representative clustering to the user, as it guarantees, that by a user specified level of confidence α, the true probability $P(X_i, \tau)$ is guaranteed to be larger than $\hat{P}(X_i, \tau, \alpha)$. To compute $\hat{P}(X_i, \tau, \alpha)$ as in Definition 5 we argue that in our setting the central limit theorem is applicable, since the sample size $|X|$ should be sufficiently large (≥ 30 as a rule of thumb [7]). Furthermore, the probability $P(X_i, \tau)$ should not be extremely small, since a cluster representative having an extremely small value of $P(X_i, \tau)$ is meaningless and should not be returned to the user in the first place. In the case where all cluster representatives have an extremely small $P(X_i, \tau)$ value, the parameter τ should be increased to obtain meaningful representatives. Yet, we note that more accurate approximations can be obtained using Wilson Score Intervals [53] or using exact binomial confidence intervals [11].

5.3 Selection of Representative Worlds

Using the techniques of Section 5.2 we can estimate, for a given τ the probability of any drawn possible world to be a τ-representative. In this section, we show how good representatives having a high confidence and low τ can be extracted automatically from a set of sampled worlds. Furthermore, when more than a single representative world is returned, a requirement is to minimize redundancy between sets of worlds represented by each representative [12, 29, 55]. This requirement is important in order to avoid overly similar clustering representatives. To solve this challenge,

we propose a general approach to first derive a clustering of the set of clusterings PC that have been obtained by applying the domain specific clustering algorithm $\mathcal{C}$ to sampled possible worlds X. Then, a single representative clustering R is chosen from each cluster of PC such that τ is minimized while the fraction of drawn possible clusterings is maximized. Formally:

DEFINITION 6 (REPRESENTATIVE WORLDS CLUSTERING). *Let PC denote the set of possible clusterings derived from sampled worlds* $X = \{X_1, ..., X_n\}$. *Let D be a* $|X| \times |X|$ *matrix such that*

$$D_{i,j} := \mathrm{dist}\,(X_i, X_j).$$

Let $\mathcal{C}'$ be a metric clustering algorithm based on dist *and let $\mathcal{C}'(PC)$ denote the meta-clustering returned by applying $\mathcal{C}'$ to the set PC of possible clusters. For each meta-cluster $Y \in \mathcal{C}'(PC)$, a* Representative Worlds Clustering *returns a triple $(R, \tau, \hat{P}(R, \tau, \alpha))$, where $R \in Y$ is the clustering chosen to represent Y, and R is an α-significant representative (cf. Definition 5) with a probability ϕ of at least $\hat{P}(R, \tau, \alpha)$.*

In Definition 6, two parameters are undefined, the choice of the clustering algorithm $\mathcal{C}'(PC)$ and a heuristic to obtain a representative from each meta-cluster in $\mathcal{C}'(PC)$. For the choice of clustering algorithm $\mathcal{C}'$, any clustering algorithm which is appropriate for general metric spaces could be used [33, 28, 34]. For the problem of defining a representative for each a meta-cluster Y, we propose the following two heuristics. Our first heuristic requires all possible clusterings in a meta-cluster $Y \in \mathcal{C}'(PC)$ to be represented.

DEFINITION 7 (COMPLETE REPRESENTATIVE). *For a meta-cluster $Y \in \mathcal{C}'(PC)$, the* complete representative *is the clustering*

$$R_{complete} := \arg\min_{R \in Y} \left(\max_{R' \in R} \left(\mathrm{dist}(R, R') \right) \right)$$

which has the minimum maximum distance

$$\tau = \max_{R' \in R} \left(\mathrm{dist}(R, R') \right)$$

to all other clusterings in Y.

This representative R_{complete} can be returned as a τ-ϕ-representative with confidence probability $\phi = \hat{P}(R_{\mathrm{complete}}, \tau, \alpha)$ using a user specified level of confidence α as described in Section 5.2.

A drawback of the complete representative approach is that the value of τ may grow arbitrarily large, being at least half of the corresponding clusters diameter. A τ-representative having an overly large τ value, such as an ARI-distance [26] value greater than 0.2, may have no semantic meaning to the user, as the space of clusterings represented by this τ-representative grows too large to allow meaningful decision making. Furthermore, a large value of τ yields overlapping clusters. For instance, for a pair of complete representatives R_i and $R_j, i \neq j$, where R_i is an τ_i-representative and R_j is an τ_j representative, it may hold that for a single sampled clustering $X_k \in X$ that $\mathcal{D}(X_k, R_i) \leq \tau_i$ and $\mathcal{D}(X_k, R_j) \leq \tau_j$. This drawback of complete representatives can be particularly bad, if the underlying clustering algorithm $\mathcal{C}$ allows clusters to have a large diameter (e.g., $\mathcal{C}$ is k-means). In contrast, complete representatives may yield good results in

Table 1: Datasets [4] and Parameters

Dataset	Tuples	Dim	ϵ	MinPts
abalone	4177	8	.08	5
ecoli	336	8	.12	4
glass	214	10	.3	4
iris	150	4	.4	4
letter	20000	16	.04	5
segmentation	2310	19	.3	4
wine	178	13	.6	4
yeast	1484	8	.08	5

settings where density-based clustering algorithms such as DBSCAN are used.

For the general case, we propose a different approach, where a maximum threshold for τ is provided. This parameter, which is specific to the chosen distance function dist, should be chosen in a way that a user should treat two clusterings, having a distance of no more than τ as similar.

DEFINITION 8 (τ_{max}-CLUSTERING). *Given a $\tau_{\max}$ threshold, for a cluster $C \in \mathcal{C}'(PC)$ a $\tau_{\max}$ representative is a τ-ϕ-representative, such that $\tau \leq \tau_{\max}$ given by*

$$R_{\tau_{\max}} := \arg\max_{R \in Y} \left(\sum_{C_i \in Y} I\left(\mathrm{dist}(R, C_i) \leq \tau_{\max}\right) \right) \cdot C_i.supp.$$

Again, this representative $R_{\tau_{\max}}$ can be returned as a τ-ϕ-representative by computing a confidence probability $\phi = \hat{P}(R_{\tau_{\max}}, \tau_{\max}, \alpha)$ with a user specified level of confidence α as described in Section 5.2.

The main drawback of τ_{max} clusterings is that large fractions of possible clusterings may not be assigned to any τ-representative. The semantics of such result, however, may be useful, indicating that a large fraction of possible clusterings deviate too much from other clusterings. This indication of high heterogeneity of possible clusterings has to be considered when making decisions based on the uncertain data set $\mathcal{DB}$.

6. EXPERIMENTS

6.1 Experimental Setup

The focus of this paper is to mitigate the effect of uncertainty by obtaining an uncertain clustering that is similar to applying algorithm $\mathcal{C}$ on the real, unknown data set oracle($\mathcal{DB}$), independent of the choice of $\mathcal{C}$.

Datasets and Ground Truth. Evaluations have been run on synthetic data as well as on the datasets summarized in Table 1. For reasons of comparability, we normalized all datsets to [0,1] in each dimension. In a preparation step, we apply a traditional (certain) clustering algorithm, DBSCAN [17], to obtain the ground-truth clustering $\mathcal{C}(\mathrm{oracle}(\mathcal{DB}))$. We then tuned the parameters ϵ and MinPts in order to yield a high F-measure for predicting the class labels of each database object. Those parameters are specified along with the datasets in Table 1.

Then, we discarded the class-information from the datasets, and treated the result of $\mathcal{C}$ as the ground truth $\mathcal{C}(\mathrm{oracle}(\mathcal{DB}))$. Recall that our goal is to compute clustering results on an

uncertain version of each dataset similar to $\mathcal{C}(\text{oracle}(\mathcal{DB}))$, independent of the quality of $\mathcal{C}$ in terms of its F-measure. Yet, the parameters of $\mathcal{C}$ should have meaningful values in our setting, to avoid effects such as having only a single cluster or no clusters at all.

Uncertainty Generation. In an uncertain setting we do not have access to the certain database oracle$(\mathcal{DB})$ and are rather given an uncertain database $\mathcal{DB}$. Thus, for each object $o \in \mathcal{DB}$, we draw a new object using a multivariate Gaussian or multivariate uniform distribution. In both cases, we use a parameter ext to describe the uncertainty. In the Gaussian case we uniformly chose a standard deviation $\sigma \in [0; ext/4]$ in each dimension and generated a center of a generating probability distribution by drawing one sample point g from the Gaussian PDF with $\mu = o$. Using g as observation of o, we generate $i-1$ additional points from the normal distribution $\mu = g$. The resulting i points correspond to samples of an uncertain object observed at location g.

In case of uniform distribution, a rectangle r was constructed having an extent chosen uniformly in the interval $[0, ext]$ in each dimension. The resulting new object u is chosen uniformly from this interval. Then, the rectangle r is centered at u and $i-1$ more points are drawn uniformly from r. In addition to o, which is guaranteed to be inside r by construction we generated $i-1$ additional points uniformly distributed in r. All generated uncertain objects form the uncertain database $\mathcal{DB}$.

For our experiments, we used $ext = 0.04$ and $i = 10$. Our approach sampled $|X| = 100$ possible worlds, assuming mutual independence of objects.

Algorithms. In our experiments we set the parameters of our framework to $\mathcal{C} = \text{DBSCAN}$ [17] and $\mathcal{C}' = \text{PAM}$ [33], a k-medoid variant, and $dist = 1 - \text{ARI}$, i.e., a distance between clusterings, based on the Adjusted Rand Index [26].

As baseline, we use a *Median Clustering* (MC) of the data set, which performs DBSCAN on the uncertain objects by reducing each uncertain object to one single possible alternative which corresponds to the median of its alternatives. This approach is a representative of naïve approaches [41, 21, 23] which reduce the uncertainty information of an uncertain object to a single point (the median in this case). A comparison partner from the literature is FDBSCAN [36]. The parameters were chosen identically for MC, FDBSCAN, and our approach.

All algorithms were obtained from or implemented in the ELKI-Framework [1] and executed on a 64-Bit Linux Computer with eight cores at 3.40GHz and 32GB RAM.

6.2 Experiments on Synthetic Data

Before evaluating the proposed approach in a broad experimental setting, we first demonstrate the difference regarding the result of the clustering task between our technique and previous work on clustering uncertain objects (represented by MC). For this purpose, we generated a toy example consisting of three Gaussian distributed point clouds $\{A, B, C\}$ which represent our ground truth data. After adding Gaussian uncertainty, as described in the previous section, all objects consist of several sample points which can be covered by a minimum bounding rectangle; these rectangles are shown in Figures 3 and 4.

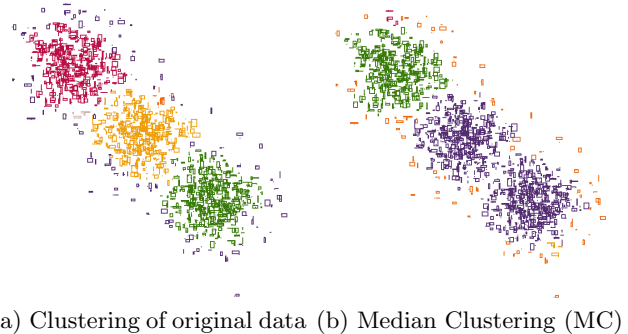Figure 3(a) illustrates the clustering of the original data set without uncertainty. Objects belonging to the same clus-

(a) Clustering of original data (b) Median Clustering (MC)

Figure 3: Clustering results of sample dataset

(a)$\hat{P}(X_1, 0.075, 0.95) = 0.36$ (b)$\hat{P}(X_2, 0.094, 0.95) = 0.25$

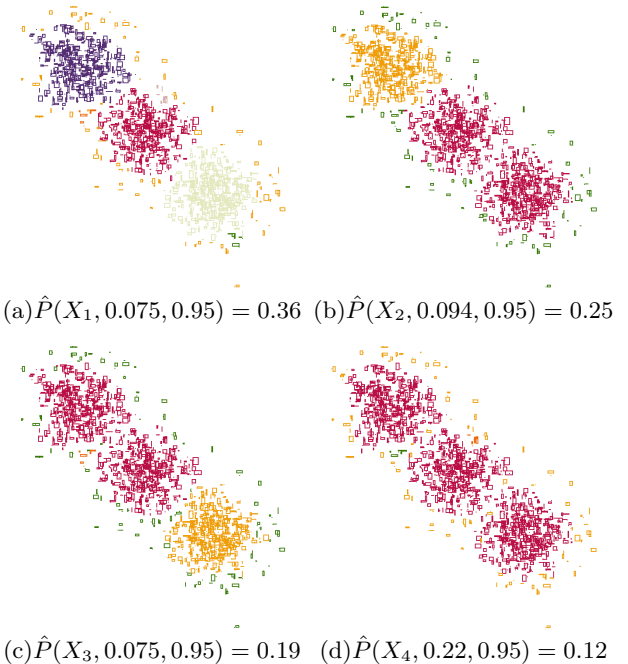(c)$\hat{P}(X_3, 0.075, 0.95) = 0.19$ (d)$\hat{P}(X_4, 0.22, 0.95) = 0.12$

Figure 4: Four representative clusterings

ter are plotted using the same color. Outliers are plotted in a different color. Figure 3(b) shows the result of MC, which yields a different clustering compared to the original one, since the lower two point clouds are merged to a single cluster.

Next consider the results of our approach when generating four representative clusterings $\{X_1, ..., X_4\}$ in Figure 4. First, note that the four results coarsely reflect the four expected possible results of a density based clustering approach ($\{A\}, \{B\}, \{C\}$), ($\{A, B\}, \{C\}$), ($\{A\}, \{B, C\}$) and ($\{A, B, C\}$). The corresponding confidence probabilities $\hat{P}(X_i, \tau, \alpha)$ (cf. Definition 5) are shown for each representative. For instance, representative X_1, shown in Figure 4(a), is an $\alpha = 0.95$-significant representative having a probability of 0.38 to have an ARI-distance of at most $\tau = 0.075$ to the ground-truth clustering $\mathcal{C}(\text{oracle}(\mathcal{DB}))$.

The real ARI-distances of the four representatives to $\mathcal{C}(\text{oracle}(\mathcal{DB}))$ are 0.038, 0.400, 0.404, and 0.851, respectively. In this toy example, the clustering with the smallest ARI to the base clustering has the highest probability. This
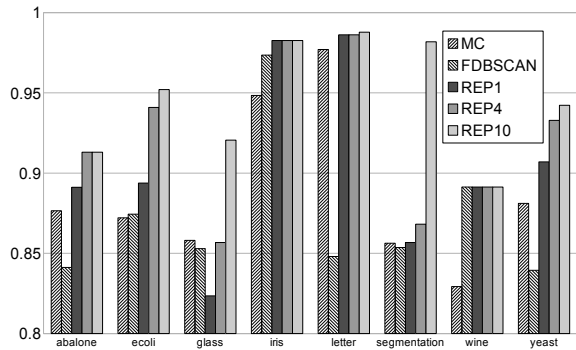
Figure 5: ARI on all datasets.

Figure 6: ARI vs. the number of representatives.

Figure 7: Confidence depending on samples.

is not always the case and our approach might also return a result having a large distance with the highest probability. Yet, our approach usually returns at least one possible clustering having a very high similarity with the true clustering. However, more importantly, unlike existing approaches, our approach is able to assess the probabilistic quality of its results and can provide multiple representative clusterings for the user to choose from.

6.3 Experiments on Real Data

Clustering Quality. Figure 5 illustrates the results under the default values on all tested datasets. Shown is the Adjusted Rand Index (ARI) [26] to the certain clustering on the original dataset. Thus, a value of one means that the method produces the same clustering result on the uncertain data than on the certain data, whereas a value closer to zero means that the two clusterings differ drastically. As observed, in several cases the rather simple MC performs better than the more sophisticated FDBSCAN. This might be because of the shortcomings of FDBSCAN regarding the consideration of possible worlds semantics as discussed in Section 2. Yet still, although the MC approach returns a clustering of a possible world, it cannot assign any measure of confidence to it, possibly resulting in a highly unlikely world. This becomes obvious when revisiting the results in Figure 5. Even for the case were only one representative is returned by our method (REP1), this representative resembles the original clustering better and in addition it also carries a confidence about its similarity to the true clustering. For instances of our method with multiple representatives, the figure shows the ARI of the representative with the minimum distance. Thus, increasing the number of representatives (REP4, REP10) ensures that at least one representative resembles the original clustering very closely.
Number of Representatives. An important question is how many representatives should be presented to the user. Presenting the user too few representatives may yield an insufficient understanding of the possible outcomes and the result may not contain a clustering which is close to the "true" clustering at all, while presenting too many representatives may overwhelm the user. In Figure 6, we show the averaged ARI over all considered datasets when increasing the number of representatives. Observe that the average ARI of all our representatives (weighted by the confidence of the representatives) decreases in comparison to the MC approach (we exclude the FDBSCAN in this graph due to its larger deviation). This can be explained by the diver-
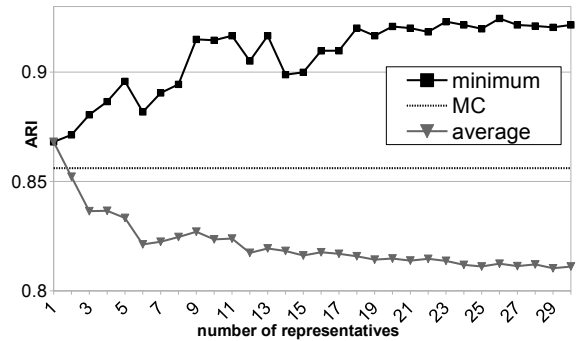
sity that increases with a higher number of representatives. On the other hand, the closest representative to the original clustering yields a higher score when increasing the number of representatives – as a larger number of representatives increases the likelihood of having a representative close to the ground truth clustering. Summarizing, our experiments show that the quality of our result gains only up to about ten representatives returned, after which no significant improval can be seen anymore. Furthermore, if becomes apparent that a set of four cluster representatives already yields fair results in most cases, while it can still be considered as concise enough to be represented to an average user.
Number of Samples. In our next experiment, we investigate how many samples $|X|$ are required in order to obtain significant results on the D31 data set [51] using the same parameter setting as for *yeast* (cf. Table 1). For this purpose, we aggregated the $\hat{P}(X_i, \tau, \alpha)$ of all cluster representatives X_i for $\tau = 0.1$, $\alpha = 0.95$ for different values of $|X|$. The result is shown in Figure 7, where it can be observed that a larger sample size $|X|$ increases the lower probability bounds obtained by Definition 5. More information on obtaining confidence intervals for a binomial probability function such as $P(X_i, \tau)$ can be found in the literature [24].
Runtime. The runtime of our approach directly corresponds to the number of samples we utilize. Thus our approach will always be slower in terms of runtime than other approaches like MC for the exchange of more valuable information and insights into the dataset, which is normal for data mining tasks such as clustering. Thus we present, in Figure 8, which modules of the process effect runtime the
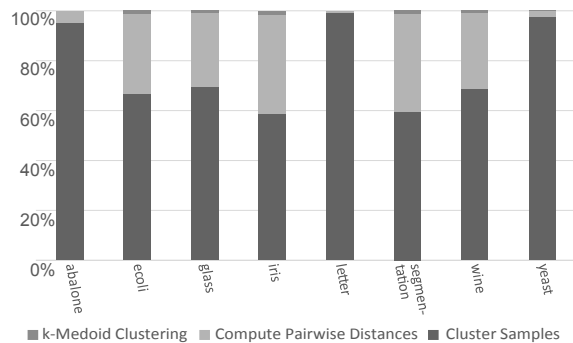
Figure 8: Relative Runtime.

most. We divided the procedure of finding cluster representatives into three steps:

- Cluster Samples: This step includes drawing possible instances of the database and clustering them using DBSCAN. This process is repeated until $|X| = 100$ instances have been processed.
- Compute Pairwise Distances: During the last step, ARI-distances between clustered database instances are required several times, thus we precomputed the $\frac{100 \cdot 99}{2}$ pairwise distances.
- k-Medoid Clustering: In this step, we perform k-medoid clustering of the sampled (certain) clusterings.

The first two steps of the overall procedure are the computational bottleneck. Obviously, applying $\mathcal{C}$ a large number of times is computationally expensive. The second step of computing the pairwise distances of the clustering results usually takes less time. This step strongly depends on the characteristics of the outcome of the first step. Specifically, computing the ARI-distance between two clusterings becomes more expensive if the clusterings contain more clusters. ARI is based on the pre-computation of the cluster contingency table which counts the number of objects in each pair of clusters of two clusterings. The number of clusters in a clustering is of course dependent on the dataset and the parameter settings.

7. CONCLUSIONS

We presented a general solution for clustering of uncertain objects. Our challenge was to develop a framework making any clustering that has been developed for certain data applicable for the case of uncertain data. We approached this challenge by employing a sampling approach to obtain a number of possible database instances from the uncertain database. Applying a domain specific clustering algorithm to each obtained database instance yields a (possibly large) set of different clusterings. Therefore, the challenge is to find a representative solution for all these possible clusterings. For this purpose, we defined the notion of τ-ϕ-representative clusterings: a τ-ϕ-representative clustering is a clustering having probability at least ϕ to have a distance of at most τ to the actual clustering of the data if the data were certain. Our solution follows a sampling approach, which returns clusterings that are guaranteed to be τ-ϕ-representative clusterings at a user specified level of significance. To the best of our knowledge, our approach is the first to yield clusterings associated with confidences, allowing the user to assess the quality of the clustering result,

and conforming to the possible worlds semantics. Furthermore, by returning multiple representative clusterings to the user, we can improve the quality (and therefore usefulness) of results, as shown by our experimental study.

8. REFERENCES

[1] E. Achtert, H.-P. Kriegel, E. Schubert, and A. Zimek. Interactive data mining with 3D-Parallel-Coordinate-Trees. In *Proc. SIGMOD*, pages 1009–1012, 2013.

[2] P. Agrawal, O. Benjelloun, A. D. Sarma, C. Hayworth, S. Nabar, T. Sugihara, and J. Widom. Trio: A system for data, uncertainty, and lineage. In *Proc. VLDB*, 2006.

[3] P. Agrawal, A. D. Sarma, J. Ullman, and J. Widom. Foundations of uncertain-data integration. *PVLDB*, 3(1-2):1080–1090, 2010.

[4] K. Bache and M. Lichman. UCI machine learning repository, 2013.

[5] T. Bernecker, H.-P. Kriegel, M. Renz, F. Verhein, and A. Züfle. Probabilistic frequent itemset mining in uncertain databases. In *Proc. KDD*, pages 119–128, 2009.

[6] J. Boulos, N. Dalvi, B. Mandhani, S. Mathur, C. Re, and D. Suciu. Mystiq: a system for finding more answers by using probabilities. In *Proc. SIGMOD*, pages 891–893, 2005.

[7] L. D. Brown, T. Cai, and A. DasGupta. Interval estimation for a binomial proportion. *Stat. Sci.*, 16(2):101–133, 2001.

[8] R. J. G. B. Campello, D. Moulavi, and J. Sander. Density-based clustering based on hierarchical density estimates. In *Proc. PAKDD*, pages 160–172. 2013.

[9] M. Chau, R. Cheng, B. Kao, and J. Ng. Uncertain data mining: An example in clustering location data. In *Proc. PAKDD*, pages 199–204, 2006.

[10] R. Cheng, S. Singh, and S. Prabhakar. U-DBMS: a database system for managing constantly-evolving data. In *Proc. VLDB*, 2005.

[11] C. Clopper and E. S. Pearson. Probable inference, the law of succession, and statistical inference. *Biometrika*, 26:404–413, 1934.

[12] Y. Cui, X. Z. Fern, and J. G. Dy. Non-redundant multi-view clustering via orthogonalization. In *Proc. ICDM*, pages 133–142, 2007.

[13] N. Dalvi and D. Suciu. Efficient query evaluation on probabilistic databases. *VLDB J.*, 16(4):523–544, 2007.

[14] X. H. Dang, I. Assent, and J. Bailey. Multiple clustering views via constrained projections. In *Proc. SIAM DM Workshop MultiClust*, 2012.

[15] A. Deshpande, C. Guestrin, S. R. Madden, J. M. Hellerstein, and W. Hong. Model-based approximate querying in sensor networks. *VLDB J.*, 14(4):417–443, 2005.

[16] T. Emrich, H.-P. Kriegel, N. Mamoulis, M. Renz, and A. Züfle. Querying uncertain spatio-temporal data. In *Proc. ICDE*, 2012.

[17] M. Ester, H.-P. Kriegel, J. Sander, and X. Xu. A density-based algorithm for discovering clusters in large spatial databases with noise. In *Proc. KDD*, pages 226–231, 1996.

[18] I. Färber, S. Günnemann, H.-P. Kriegel, P. Kröger, E. Müller, E. Schubert, T. Seidl, and A. Zimek. On using class-labels in evaluation of clusterings. In *Proc. ACM SIGKDD Workshop MultiClust*, 2010.

[19] G. Ghinita, P. Karras, P. Kalnis, and N. Mamoulis. Fast data anonymization with low information loss. In *VLDB*, pages 758–769, 2007.

[20] D. Gondek and T. Hofmann. Non-redundant clustering with conditional ensembles. In *Proc. KDD*, pages 70–77, 2005.

[21] F. Gullo, G. Ponti, and A. Tagarelli. Clustering uncertain data via k-medoids. In *Scalable Uncertainty Management*, pages 229–242, 2008.

[22] F. Gullo, G. Ponti, and A. Tagarelli. Minimizing the variance of cluster mixture models for clustering uncertain objects. *Stat. Anal. Data Min.*, 6(2):116–135, 2013.

[23] F. Gullo and A. Tagarelli. Uncertain centroid based partitional clustering of uncertain data. *PVLDB*, 5(7):610–621, 2012.

[24] W. Hoeffding. Probability inequalities for sums of bounded random variables. *JASA*, 58(301):13–30, 1963.

[25] J. Huang, L. Antova, C. Koch, and D. Olteanu. Maybms: A probabilistic database management system. In *Proc. SIGMOD*, pages 1071–1074, 2009.

[26] L. Hubert and P. Arabie. Comparing partitions. *J. Classif.*, 2(1):193–218, 1985.

[27] A. K. Jain. Data clustering: 50 years beyond k-means. *Pattern Recognition Lett.*, 31(8):651–666, 2010.

[28] A. K. Jain, M. N. Murty, and P. J. Flynn. Data clustering: A review. *ACM CSUR*, 31(3):264–323, 1999.

[29] P. Jain, R. Meka, and I. S. Dhillon. Simultaneous unsupervised learning of disparate clusterings. *Stat. Anal. Data Min.*, 1(3):195–210, 2008.

[30] R. Jampani, F. Xu, M. Wu, L. Perez, C. Jermaine, and P. J. Haas. The monte carlo database system: Stochastic analysis close to the data. *ACM Trans. Database Syst.*, 36(3):18:1–18:41, 2011.

[31] B. Jiang, J. Pei, Y. Tao, and X. Lin. Clustering uncertain data based on probability distribution similarity. *IEEE TKDE*, 25(4):751–763, 2013.

[32] B. Kao, S. D. Lee, D. W. Cheung, W. S. Ho, and K. F. Chan. Clustering uncertain data using voronoi diagrams. In *Proc. ICDM*, pages 333–342, 2008.

[33] L. Kaufman and P. J. Rousseeuw. *Finding Groups in Data: An Introduction to Cluster Analyis*. John Wiley&Sons, 1990.

[34] H.-P. Kriegel, P. Kröger, J. Sander, and A. Zimek. Density-based clustering. *WIREs DMKD*, 1(3):231–240, 2011.

[35] H.-P. Kriegel, P. Kröger, and A. Zimek. Subspace clustering. *WIREs DMKD*, 2(4):351–364, 2012.

[36] H.-P. Kriegel and M. Pfeifle. Density-based clustering of uncertain data. In *Proc. KDD*, pages 672–677, 2005.

[37] H.-P. Kriegel and M. Pfeifle. Hierarchical density-based clustering of uncertain data. In *Proc. ICDM*, pages 689–692, 2005.

[38] S. D. Lee, B. Kao, and R. Cheng. Reducing uk-means to k-means. In *ICDM Workshops*, pages 483–488, 2007.

[39] J. Li, B. Saha, and A. Deshpande. A unified approach to ranking in probabilistic databases. *PVLDB*, 2(1):502–513, 2009.

[40] J. MacQueen. Some methods for classification and analysis of multivariate observations. In *5th Berkeley Symposium on Mathematics, Statistics, and Probabilistics*, volume 1, pages 281–297, 1967.

[41] W. K. Ngai, B. Kao, C. K. Chui, R. Cheng, M. Chau, and K. Y. Yip. Efficient clustering of uncertain data. In *Proc. ICDM*, pages 436–445, 2006.

[42] J. Pei, M. Hua, Y. Tao, and X. Lin. Query answering techniques on uncertain and probabilistic data: tutorial summary. In *Proc. SIGMOD*, pages 1357–1364, 2008.

[43] A. D. Sarma, O. Benjelloun, A. Halevy, and J. Widom. Working models for uncertain data. In *Proc. ICDE*, 2006.

[44] P. Sen, A. Deshpande, and L. Getoor. Prdb: Managing and exploiting rich correlations in probabilistic databases. *VLDB J.*, 18(5):1065–1090, 2009.

[45] K. Sim, V. Gopalkrishnan, A. Zimek, and G. Cong. A survey on enhanced subspace clustering. *Data Min. Knowl. Disc.*, 26(2):332–397, 2013.

[46] M. A. Soliman, I. F. Ilyas, and K. C.-C. Chang. Top-k query processing in uncertain databases. In *Proc. ICDE*, pages 896–905, 2007.

[47] A. Struyf, M. Hubert, and P. Rousseeuw. Clustering in an object-oriented environment. *Journal of Statistical Software*, 1(4):1–30, 1997.

[48] L. Sun, R. Cheng, D. W. Cheung, and J. Cheng. Mining uncertain data with probabilistic guarantees. In *Proc. KDD*, pages 273–282, 2010.

[49] Y. Tao, R. Cheng, X. Xiao, W. K. Ngai, B. Kao, and S. Prabhakar. Indexing multi-dimensional uncertain data with arbitrary probability density functions. In *Proc. VLDB*, pages 922–933, 2005.

[50] G. Trajcevski, R. Tamassia, P. Scheuermann, D. Hartglass, and C. Zamierowski. Ranking continuous nearest neighbors for uncertain trajectories. *VLDB J.*, 20(5):767–791, 2011.

[51] C. J. Veenman, M. J. T. Reinders, and E. Backer. A maximum variance cluster algorithm. *IEEE TPAMI*, 24(9):1273–1280, 2002.

[52] S. Wallis. Binomial confidence intervals and contingency tests: Mathematical fundamentals and the evaluation of alternative methods. *Journal of Quantitative Linguistics*, 20(3):178–208, 2013.

[53] E. B. Wilson. Probable inference, the law of succession, and statistical inference. *JASA*, 22:209–212, 1927.

[54] K. Yi, F. Li, G. Kollios, and D. Srivastava. Efficient processing of top-k queries in uncertain databases. In *Proc. ICDE*, 2008.

[55] A. Zimek and J. Vreeken. The blind men and the elephant: On meeting the problem of multiple truths in data from clustering and pattern mining perspectives. *Mach. Learn.*, 2013.