Journal of Intelligent Information Systems
https://doi.org/10.1007/s10844-020-00624-7

Uncertain distance-based outlier detection with arbitrarily shaped data objects

Fabrizio Angiulli[1] · **Fabio Fassetti**[1]

Received: 7 February 2020 / Revised: 24 September 2020 / Accepted: 24 September 2020 /
Published online: 15 October 2020

Abstract
Enabling information systems to face anomalies in the presence of uncertainty is a compelling and challenging task. In this work the problem of unsupervised outlier detection in large collections of data objects modeled by means of arbitrary multidimensional probability density functions is considered. We present a novel definition of *uncertain distance-based outlier* under the attribute level uncertainty model, according to which an uncertain object is an object that always exists but its actual value is modeled by a multivariate pdf. According to this definition an uncertain object is declared to be an outlier on the basis of the expected number of its neighbors in the dataset. To the best of our knowledge this is the first work that considers the unsupervised outlier detection problem on data objects modeled by means of arbitrarily shaped multidimensional distribution functions. We present the UDBOD algorithm which efficiently detects the outliers in an input uncertain dataset by taking advantages of three optimized phases, that are parameter estimation, candidate selection, and the candidate filtering. An experimental campaign is presented, including a sensitivity analysis, a study of the effectiveness of the technique, a comparison with related algorithms, also in presence of high dimensional data, and a discussion about the behavior of our technique in real case scenarios.

Keywords Nearest neighbors · Outlier detection · Uncertain data · Unsupervised learning

1 Introduction

Traditional data analysis techniques deal with feature vectors having *deterministic* values. Thus, data *uncertainty* is usually ignored in the problem formulation. However, *uncertainty*

A preliminary version of this work appears in Angiulli and Fassetti (2013).

✉ Fabrizio Angiulli
fabrizio.angiulli@unical.it

Fabio Fassetti
fabio.fassetti@unical.it

[1] DIMES, University of Calabria, 87036, Rende, CS, Italy

arises in real data in many ways, since the data may contain errors or may be only partially complete (Lindley 2006). The uncertainty may result from the limitations of the equipment, indeed physical devices are often imprecise due to *measurement errors*. Another source of uncertainty are *repeated measurements*, e.g. sea surface temperature could be recorded multiple times during a day. Also, in some applications data values are *continuously changing*, as positions of devices or observations associated with natural phenomena, and these quantities can be represented by using an uncertain model.

Simply disregarding uncertainty may led to less accurate conclusions or even inexact ones. This has created a need for uncertain data management techniques (Aggarwal and Yu 2009) managing data records typically represented by probability distributions (Mohri 2003; Kriegel and Pfeifle 2005; Bi and Zhang 2004; Aggarwal and Yu 2008; Angiulli and Fassetti 2012, 2007; Aggarwal 2014; Khan et al. 2018), In this work it is assumed that an *uncertain object* is an object that always exists but its actual value is uncertain and modeled by a multivariate probability density function. This notion of uncertain object has been extensively adopted in the literature and corresponds to the *attribute level uncertainty model* viewpoint (Green and Tannen 2006).

In particular, we deal with the problem of *detecting outliers in uncertain data*. An *outlier* is an observation that differs so much from others as to arouse suspicion that it was generated by a different mechanism (Hawkins 1980). As a major contribution, we introduce a definition of uncertain outlier representing the generalization of the classic distance-based outlier definition (Knorr et al. 2000; Ramaswamy et al. 2000; Angiulli and Pizzuti 2005; Angiulli et al. 2006) to the management of uncertain data modeled as arbitrary probability density functions. The distance-based definition is a solid one: it has been introduced in order to overcome some limitations of statistical definitions, generalizes the notion of outlier provided by several discordance tests developed in statistics, is suitable for multivariate data, and can be applied even if the distribution of the data is unknown. The contributions of the work are summarized next.

- To the best of our knowledge, this is the first unsupervised outlier detection technique working on data objects modeled by means of arbitrarily shaped multidimensional distribution functions.
- We introduce a novel definition of uncertain outlier representing the generalization of the classic distance-based outlier definition (Knorr et al. 2000; Ramaswamy et al. 2000; Angiulli and Pizzuti 2005) to the management of uncertain data modeled as pdfs.
- Our approach consists in declaring an object as an outlier if the probability that it has at least k close neighbors is low. Hence, it corresponds to perform a nearest neighbor density estimate on all the possible dataset outcomes. As such, its semantics is completely different from previously introduced unsupervised approaches for outlier detection on uncertain data (Aggarwal and Yu 2008; Wang et al. 2009; Jiang and Pei 2011).
- We show how the decision rule associated with the here introduced definition, although difficult to compute, can be truthfully implemented.
- We provide an efficient uncertain distance-based outlier detection algorithm working on any domain and with any distance function.

The rest of the paper is organized as follows. Section 2 introduces the notion of distance-based uncertain outlier. Section 3 discusses work related to the one here presented. Section 4 shows how to compute the outlier probability. Section 5 presents the outlier detection method. Section 6 illustrates experimental results. Finally, Section 7 concludes the work.

2 Preliminaries

2.1 Uncertain objects

Let ($\mathbb{D}$, dist) denote a metric space, where $\mathbb{D}$ is a set, also called *domain*, and dist is a *metric distance* on $\mathbb{D}$. (e.g., $\mathbb{D}$ is the d-dimensional real space $\mathbb{R}^d$ equipped with the Euclidean distance dist).

A *certain object* v is an element of $\mathbb{D}$. An *uncertain object* x is a random variable having domain $\mathbb{D}$ with associated probability density function f^x, where $f^x(v)$ denotes the density of x in v. We note that a certain object v can be regarded as an uncertain one whose associated pdf f^v is $\delta_v(u)$, where $\delta_v(u) = \delta(\mathbf{0})$, for $u = v$, and $\delta_v(u) = \mathbf{0}$, otherwise, with $\delta(u)$ denoting the Dirac delta function.

Given a set $S = \{x_1, \ldots, x_N\}$ of uncertain objects, an *outcome* I_S of S is a set $\{v_1, \ldots, v_N\}$ of certain objects such that $f^{x_i}(v_i) > 0$ $(1 \leq i \leq N)$. The pdf f^S associated with S is

$$f^S(v_1, \ldots, v_N) = \prod_{i=1}^{N} f^{x_i}(v_i).$$

Given two uncertain objects x and y, dist(x, y) denotes the continuous random variable representing the *distance* between x and y.

In the following we assume that with each object x it is given a finite region SUP(x) such that $Pr(x \notin \text{SUP}(x)) \leq \omega$ for a specific threshold ω. For example, SUP could be defined as an hyper-ball or an hyper-rectangle (e.g. the minimum bounding rectangle or MBR). If x has finite support, the threshold ω can be always set to 0. Note that under the above assumption the error involved in the calculation of the probability $Pr(\text{dist}(x, y) \leq R)$, with x and y two uncertain objects, is the square of ω.

The *minimum distance* $mindist(x, y)$ between uncertain objects x and y is defined as $\min\{\text{dist}(u, v) : u \in \text{SUP}(x) \text{ and } v \in \text{SUP}(y)\}$, while the *maximum distance* $maxdist(x, y)$ between x and y is defined as $\max\{\text{dist}(u, v) : u \in \text{SUP}(x) \text{ and } v \in \text{SUP}(y)\}$.

2.2 Uncertain outliers

Given an uncertain dataset **DS**, $D_k(x, \mathbf{DS})$ (or $D_k(x)$, for short) denotes the continuous random variable representing the distance between x and its k-th nearest neighbor in $\mathbf{DS} \setminus \{x\}$. Next we define the notion of *outlier in an uncertain dataset*. For the sake of brevity, in the sequel, we will refer to an outlier in an uncertain dataset as to an *uncertain outlier*.

Definition 1 *Given an uncertain dataset **DS**, an uncertain distance-based outlier in **DS** according to parameters k, R and $\delta \in (0, 1)$ is an uncertain object x of **DS** such that the following relationship holds:*

$$Pr(D_k(x, \mathbf{DS}) \leq R) \leq 1 - \delta.$$

That is to say, an uncertain distance-based outlier is a dataset object for which the probability of having k dataset objects besides itself within distance R is smaller than $1 - \delta$.

Let N be the number of objects in **DS**. In order to determine the probability $D_k(x)$, the

following multi-dimensional integral has to be computed, where $\mathbf{DS}'$ denotes the uncertain dataset $\mathbf{DS} \setminus \{x\}$ and $I_{\mathbf{DS}'}$ a generic outcome of $\mathbf{DS}'$ (see also Section 2.1):

$$\int_{\mathbb{D}^N} f^x(v) \cdot f^{\mathbf{DS}'}(I_{\mathbf{DS}'}) \cdot \mathbf{I}[D_k(v, I_{\mathbf{DS}'}) \leq R] \, \mathrm{d}I_{\mathbf{DS}'} \, \mathrm{d}v, \tag{1}$$

where the function $\mathbf{I}(\cdot)$ outputs 1 if the probability of its argument is 1, and 0 otherwise. According to the above formulation, deciding if an object is an uncertain distance-based outlier requires to compute an integral involving all the outcomes of the dataset.

3 Related work

There exist several approaches to detect outliers in the certain setting, namely statistical-based (Davies and Gather 1993; Barnett and Lewis 1994), deviation-based (Arning et al. 1996), distance-based (Knorr et al. 2000), density-based (Breunig et al. 2000; Papadimitriou et al. 2003), reverse nearest-neighbor-based (Angiulli 2020), isolation-based (Liu et al. 2012), subspace-based (Knorr and Ng 1999; Aggarwal and Yu 2001a; Angiulli et al. 2009, 2013), knowledge-based (Angiulli and Fassetti 2014), neural network-based (Hawkins et al. 2002), support vector machine-based (Tax and Duin 2004), and many others (Chandola et al. 2009; Aggarwal 2016). Among these approaches, distance-based outlier detection methods have been shown to be effective in various scenarios (Knorr et al. 2000; Bay and Schwabacher 2003; Ghoting et al. 2006; Tao et al. 2006; Angiulli and Fassetti 2009). However, none of these techniques is designed to handle uncertain data and, as far as the uncertain setting is concerned, only a few approaches have been proposed (Aggarwal and Yu 2008; Wang et al. 2009; Jiang and Pei 2011).

The method described in Aggarwal and Yu (2008) is a density based approach designed for uncertain objects which aims at selecting outliers in subspaces. The idea of the method is to approximate the density of the dataset by means of kernel density estimation and then to declare an uncertain object as an outlier if there exists a subspace such that the probability that the object lies in a sufficiently dense region is negligible. Differently from our approach, in Aggarwal and Yu (2008) the density estimate does not take directly into account the form of the pdfs associated with uncertain objects, since it is performed by using equi-bandwidth Gaussian kernels centered in the means of the object distributions. Pdfs are then taken into account to determine the objects lying in regions of low density, where the density is computed as before mentioned. Furthermore, since the method is interested in exploring subspaces (we recall that our goal is to detect outliers in the full feature space), pdfs are always expressed as the product of d independent one-dimensional pdfs, where d is the dimension of the space, while we are able to manage arbitrarily shaped multidimensional density functions.

In Wang et al. (2009) authors present a distance-based approach to detect outliers which adopts a completely different model of uncertainty than our, that is the existential uncertainty model, according to which an uncertain object x assumes a specific value v_x with a fixed probability p_x and does not exist with probability $1 - p_x$. According to this approach, uncertain objects are not modeled by means of distribution functions, but rather are deterministic values that may either occur or not occur in an outcome of the dataset. Hence, although (Wang et al. 2009) deals with distance-based outliers, their scenario is completely different from our, and the two methods are not comparable at all.

In Jiang and Pei (2011) an uncertain object consists of a pair (l, r), where l is a tuple on a set of conditioning attributes and r is a set of tuples on a set of dependent attributes,

also called instances. To each instance $r_j \in r$ a measure of normality is assigned, consisting in the probability of observing r_j given that both r and l have been observed. The normality of an object is then obtained as the geometric mean of the normality of all its instances. Authors exploits kernel density estimation and Bayesian inference to solve their problem. Outlier instances are detected by comparing against normal ones. Outlier objects are then detected as those objects most of whose instances are abnormal. We notice that the approach presented in Jiang and Pei (2011) essentially aims at detecting the abnormal instances, that, loosely speaking, are the abnormal outcomes of the uncertain objects. Thus, the task on interest in Jiang and Pei (2011) is not comparable to that considered here. Moreover, uncertain objects are modeled in a way which is completely different from that considered here.

The work (Liu et al. 2013) describes a SVDD-based outlier detection technique on uncertain data. The approach assigns a confidence score to each example, which indicates the likelihood of an example tending normal class, and then incorporates these confidence scores into the SVDD training phase for outlier detection. Hence, the technique does not directly manage uncertain objects, but rather attempts to mitigate possible error measurements by reducing the contribution on the construction of the decision boundary of the examples with the least confidence score.

4 Outlier probability

In this section we show how the value of $Pr(D_k(x) \leq R)$ can be computed, for x a generic uncertain object of **DS**. Given a certain object v and an uncertain object y, let $p_v^y(R) = Pr(\text{dist}(v, y) \leq R)$ denote the cumulative density function representing the relative likelihood for the distance between objects v and y to assume value less or equal than R, that is

$$p_v^y(R) = Pr(\text{dist}(v, y) \leq R) = \int_{\mathcal{B}_R(v)} f^y(u)\, \mathrm{d}u, \tag{2}$$

where $\mathcal{B}_R(v)$ denotes the hyper-ball having radius R and centered in v.

Let v be an outcome of the uncertain object x. For $k \geq 1$, the probability $Pr(D_k(v, \mathbf{DS} \setminus \{x\}) \leq R)$ that v has at least k other dataset objects within distance R can be expressed as:

$$1 - \left(\sum_{S \subseteq \mathbf{DS}: |S| < k} \left(\prod_{z \in S} p_v^z(R) \cdot \prod_{z \in \mathbf{DS} \setminus S} (1 - p_v^z(R)) \right) \right), \tag{3}$$

that is one minus the probability that less than k dataset objects lie within distance R from v. Thus,

$$Pr(D_k(x) \leq R) = \int_{\mathbb{D}} f^x(v) \cdot Pr(D_k(v, \mathbf{DS} \setminus \{x\}) \leq R)\, \mathrm{d}v, \tag{4}$$

that is to say, loosely speaking, the summation over all the outcomes v of x of the occurrence probability of v multiplied by the probability that v has at least k objects within distance R over all the outcomes of the remaining dataset objects.

The subsequent section describes the algorithm UDBOD, whose aim is to quickly detect the dataset objects for which the right hand side of (4) is smaller than the provided probability threshold $1 - \delta$. Next it is discussed how to compute $p_v^y(R)$.

4.1 Computing the probability $p_v^y(R)$

Probability values $p_v^y(R)$ depend on the objects v and y, and on the real value R and involve the computation of one integral with domain of integration $\mathbb{D}$ (more precisely, the hyper-ball $\mathcal{B}_R(v)$). It is known (Lepage 1978) that given a function g, if m points $w_1, w_2, \ldots, w_m$ are randomly selected according to a given pdf f, then the following approximation holds:

$$\int g(u)\,\mathrm{d}u \approx \frac{1}{m}\sum_{i=1}^{m}\frac{g(w_i)}{f(w_i)}. \tag{5}$$

Thus, in order to compute the value $p_v^y(R)$ reported in (2), the function $g_v^y(u)$ such that $g_v^y(u) = f^y(u)$ if $\mathrm{dist}(v, u) \leq R$, and $g_v^y(u) = 0$ otherwise, can be integrated by evaluating the formula in (5) with m points w_i randomly selected according to the pdf f^y. This procedure reduces to computing the relative number of sample points w_i lying at distance not greater than R from v, that is

$$p_v^y(R) = \frac{|\{w_i : \mathrm{dist}(v, w_i) \leq R\}|}{m}. \tag{6}$$

5 Uncertain distance-based outlier detector

In this section we describe the algorithm UDBOD (for *Uncertain Distance-Based Outlier Detector*) that mines the distance-based outliers in an uncertain dataset **DS** consisting of N objects.

Definition 1 makes use of three parameters, that are k (or, equivalently, $\varrho \in (0, 1)$, by setting $k = \varrho N$), R, and δ. We point out that these parameters can be held fixed to the default values in order to perform a meaningful analysis, as experimental results show that outlier detection is little sensitive to the values of user-specific parameters. Specifically, according to the statistical and distance-based outlier detection literature (Knorr et al. 2000; Angiulli and Fassetti 2009), meaningful values for the parameter ϱ are in the range $(0, 2‰]$ (the value 1‰ is employed by default), while δ being a threshold level can be conveniently set in the range $[0.8, 0.9]$ (the value 0.9 is employed by default). As for the value of the parameter R, it will be automatically determined by UDBOD once the percentage α of outliers to detect has been specified. The value α is much more easy to determine than R and can be conveniently set to the 3‰ (Angiulli and Fassetti 2009).

Other than the above *external* parameters, the method requires some *internal* parameters, described in the sequel of the section, that do not require to be set by the user, since their optimal values are automatically determined from the external ones. Table 1 summarizes some of the symbols employed in this section, and meaningful ranges and recommended values for the parameters.

Table 1 Symbols employed in Section 5.1

Symbol	Range	Recommended value	Description
δ	0.8/0.9	0.9	Outlier probability threshold
ϱ	1‰ / 2‰	1‰	Relative number of nearest neighbors
k	$\lceil \varrho N \rceil$	$\lceil \varrho N \rceil$	Absolute number of nearest neighbors
α	[2‰, 5‰]	3‰	Percentage of outliers to detect
ϵ	1‰/2‰	2‰/1‰	Estimation error for $\hat{\alpha}$
$1-\lambda$	[0.8, 0.9]	0.8/0.9	Estimation error upper bound
s	See (8)	1,228/8,093*	Sample size for R estimation
β	[0, 1]	0.5 (See Section 6)	Mass factor
R	See Algorithm 2	See Algorithm 2	Outlier radius

*The default value is $s = 1{,}228$ ($\epsilon = 2‰$ and $1-\lambda = 0.8$) for $N < 100{,}000$, and $s = 8{,}093$ ($\epsilon = 1‰$ and $1-\lambda = 0.9$) otherwise

The pseudo-code of UDBOD is reported in Algorithm 1. It consists of three phases: parameter estimation, candidate selection, and candidate filtering.

Algorithm 1: *UDBOD.*

Input: uncertain dataset **DS**
parameter ϱ (1‰ by default)
percentage of outliers α (3‰ by default)
probability threshold δ (0.9 by default)
Output: uncertain outliers $Outliers$
// Parameter estimation phase
Determine the value for the parameter R by means of Algorithm 2
// Candidate selection phase
Determine the set $OutCands$ of candidate outliers by detecting objects x such that $D_k^1(x) > R$
// Candidate filtering phase
Set $Outliers$ to the empty set
foreach x *in* $OutCands$ **do**
  if $D_k^0(x) > R$ **then**
    Insert x into $Outliers$;
  else
    if $Pr(D_k(x) \leq R) \leq 1-\delta$ **then**
      Insert x into $Outliers$;
return the set $Outliers$

5.1 Parameter estimation phase

The *Parameter estimation phase* determines the right value R^* for the outlier radius R as a function of ϱ and α. Note that the effectiveness of the uncertain distance-based definition relies on the right selection of the radius value. Setting a meaningful value for the parameter R is a difficult task since its right value heavily depends on the characteristics of the input data. In particular, we will map the problem of setting R to the problem of setting a parameter $\beta \in [0, 1]$, by means of which the expected fraction of outliers can be controlled in a very simple and meaningful way. Indeed, as made clearer next, for $\beta = 0$ ($\beta = 1$,

resp.) we have the statistical guarantee that a subset (superset, resp.) of the actual outliers is retrieved. In order to provide the above statistical guarantees, the meaningfulness of the outlier radius is related to the number of outliers estimated by means of a sampling procedure. The following definition is preliminarily needed.

Definition 2 *Given two uncertain objects x and y, and a value $\beta \in [0, 1]$, also called mass factor, let $dist^{\beta}(x, y)$ denote the distance value:*

$$dist^{\beta}(x, y) = \beta \cdot maxdist(x, y) + (1 - \beta) \cdot mindist(x, y).$$

Note that for $\beta = 1$ the distance $\text{dist}^{\beta}(x, y)$ coincides with $maxdist(x, y)$ and that for $\beta = 0$ the distance $\text{dist}^{\beta}(x, y)$ coincides with $mindist(x, y)$, while for $\beta \in (0, 1)$, $\text{dist}(x, y)$ assumes an intermediate value.

Let $D_k^{\beta}(x, \mathbf{DS})$ (or $D_k^{\beta}(x)$, whenever the dataset **DS** is clear from the context) denote the k-th nearest neighbor distance in $\mathbf{DS} \setminus \{x\}$ according to dist^{β}.

Let α denote the percentage of outliers to be detected. Then, once the parameter $k = \lceil \varrho N \rceil$ has been fixed, the value R^* for the parameter R such that the α percent of the dataset objects has less than k objects at distance dist^{β} less than R^* can be estimated by means of the method reported in Algorithm 2.

Algorithm 2: *Parameter estimation phase.*

Input: uncertain dataset **DS**
parameter ϱ (1‰ by default)
percentage of outliers α (3‰ by default)
sample size s (8,093 by default)
mass factor β (0.5 by default)
Output: parameter R

1 Pick a random sample **RS** of s objects from **DS**
2 **foreach** *object x in **RS*** **do**
3 Compute the value $d_x = D^{\beta}_{\lceil \varrho s \rceil}(x, \mathbf{RS})$
4 Determine the value R^* such that exactly $\lceil \alpha s \rceil$ objects x of **RS** have d_x greater than R^*
5 **if** $R^* < \mu_{d_x} + 4\sigma_{d_x}$ **then**
// Estimation Correction
6 Set R^* to the smallest d_x value which is greater than $\mu_{d_x} + 4\sigma_{d_x}$;
7 **return** R^*

In order the above method to be effective, a meaningful value for the sample size s must be employed. Now, it is shown how to set the size s of the sample in order to have a statistical guarantee that the actual percentage $\widehat{\alpha}$ of objects in the whole dataset **DS** having D_k^{β} greater than the R^* is close to α.

With this aim, the following relation must hold

$$Pr(|\widehat{\alpha} - \alpha| \leq \epsilon) > 1 - \lambda, \tag{7}$$

asserting that the probability that the *estimation error*, that is the difference between $\widehat{\alpha}$ and α, is lower than an error threshold ϵ, is greater than $1 - \lambda$. Clear enough, the lower ϵ and λ, the closer $\widehat{\alpha}$ to α. By the *Central Limit* theorem, if the sample size s is large enough, then the following relationship holds:

$$Pr\left(|\widehat{\alpha} - \alpha| \leq \epsilon\right) \approx 2 \cdot \Phi\left(\frac{\epsilon\sqrt{s}}{\sqrt{\alpha(1-\alpha)}}\right) - 1.$$

Hence, the relation in (7) is satisfied if

$$s > \frac{\alpha(1-\alpha)}{\epsilon^2}\left(\Phi^{-1}\left(1-\frac{\lambda}{2}\right)\right)^2. \tag{8}$$

For example, let $\alpha = 3‰$, and let $\epsilon = 2‰$ and $\lambda = 0.2$, so that the number of uncertain outliers is between the 1‰ and the 5‰ with probability 0.8. By using (8) the sample size is $s = 1{,}228$. Now, we prove that the radius R^* returned by the *Parameter estimation phase* is meaningful for the Definition 1. First, some properties of D_k^β are introduced.

Property 1 Let x be an uncertain object for which $D_k^1(x)$ is less or equal than R. Then x is not an outlier.

Indeed, if the condition of the statement is true, then each outcome of x has at least k neighbors within radius R in every outcome of the dataset.

Property 2 Let x be an uncertain object for which $D_k^0(x)$ is greater than R. Then x is an outlier.

Indeed, if the condition of the statement is true, then each outcome of x has less than k neighbors within radius R in every outcome of the dataset. Thus, given radius R', it follows from Proposition 1 that the uncertain objects x of **DS** satisfying $D_k^1(x) > R'$ are a superset of the outliers in **DS** for $R = R'$. Moreover, it follows from Proposition 2 that the uncertain objects x of **DS** satisfying $D_k^0(x) > R'$ are a subset of the outliers in **DS** for $R = R'$.

Theorem 1 Let R_1 (R_0, resp.) be the smallest radius such exactly αN dataset objects x satisfy the condition $D_k^1(x) > R_1$ ($D_k^0(x) > R_0$, resp.), and let n_1 (n_0, resp.) the actual number of uncertain distance-based outliers in **DS** for $R = R_1$ ($R = R_0$, resp.). Then, the expected number $n = \alpha N$ of outliers in **DS** is lower bounded by n_1 (upper bounded by n_0, resp.), that is $n_1 \leq n \leq n_0$.

Proof As already pointed out, the αN objects x satisfying condition $D_k^1(x) > R_1$ are a superset of the actual number n_1 of outliers for $R = R_1$ and, consequently, $n_1 \leq \alpha N = n$. Moreover, the αN objects x satisfying condition $D_k^0(x) > R_0$ are a subset of the actual number n_0 of outliers for $R = R_0$ and, consequently, $n_0 \geq \alpha N = n$. □

This makes clear the motivation underlying the introduction of the parameter β: by properly tuning the value of β the actual number of outliers (and also of candidate outliers; see in the following) can be controlled in a very simple way. As for the value to assign to β, in the section devoted to experimental results it will be shown that $\beta = 0.5$ is a good option.

If at the expected outlier level α there is not a clear separation between the radius associated with outliers and that associated with inliers, then it can be concluded that there are less than αN true outliers in the dataset. So, in this case the fraction α should be lowered, for otherwise a considerable fraction of dataset objects would be recognized as outliers. This can be accomplished by properly lowering the radius R^*. In particular, Algorithm 2 guarantees that the computed radius R^* is at least four standard deviations far apart from the mean of the distribution of distances between sampled objects and their $\lceil \varrho s \rceil$-th nearest neighbor in the sample. Specifically, this *estimation correction* selects the smallest radius associated with objects in the sample which is not smaller than the above mentioned threshold.

5.2 Candidate selection phase

The *Candidate selection phase* fast determines the set $OutCands$ of candidate outliers by exploiting a deterministic lower bound property based on the $maxdist$ distance between uncertain objects. We start by recalling the definition of a distance-based outlier in the context of certain datasets (Knorr et al. 2000).

Definition 3 *Given a dataset of objects on which is defined a distance dist, a positive integer k and a positive real number R, an object v is said to be a (certain) distance-based outlier according to parameters k and R, if less than k objects of* ***DS*** *lie within distance R from v.*

The following result bridges the link between certain and uncertain distance-based outliers.

Theorem 2 For each δ, if x is an uncertain distance-based outliers of **DS** according to parameters k, R and δ then x is a certain distance-based outlier of **DS** for the distance $maxdist$ according to parameters k and R.

Proof We prove that if x is not a certain distance-based outliers of **DS** then x is not an uncertain distance-based outlier of **DS**. First, we notice that x is not a certain distance-based outlier according to parameters k and R if and only if the distance to its k-th nearest neighbor is smaller than R. Moreover, we recall that $D_k^1(x)$ denotes the distance from x and its k-th nearest neighbor according to the distance $maxdist$. The proof follows by Property 1. □

From the above theorem a suitable set $OutCands$ of uncertain *candidate outliers* can be obtained by regarding **DS** as a set of certain objects equipped with the certain distance $maxdist$ and by computing the certain distance-based outliers therein contained.

As an important property, next it is shown that if the employed distance function dist is a metric, then the maximum distance function $maxdist$ induced on dist is a metric as well.

Theorem 3 Let dist be a metric. Then the $maxdist$ function induced by the distance dist is a metric.

Proof Four properties have to be proven: non-negativity, symmetry, identity of indiscernibles, and triangle inequality. The first two properties immediately follows from the fact that dist is a metric.

As for the identity of indiscernibles, assume that $maxdist(x, y) = 0$, then it is the case that for each realization u of $\widehat{x}$ and v of $\widehat{y}$ such that $Pr[\widehat{x} = u \wedge \widehat{y} = v] > 0$, $\text{dist}(u, v) = 0$. Hence, by the fact that dist is a metric, $u = v$, and x and y must be the same uncertain object. As for the reverse direction, since x and y are the same random variable, u and v are always identical.

As for triangle inequality, given three generic uncertain objects x, y, and z, the triangle inequality is satisfied, that is to say that: $maxdist(x, z) + maxdist(z, y) \geq maxdist(x, y)$. Let x_1 and z_1 (y_2 and z_2, resp.) the outcomes of the uncertain objects x and z (y and z, resp.) for which the relationship $\text{dist}(x_1, z_1) = maxdist(x, z)$ ($\text{dist}(z_2, y_2) = maxdist(z, y)$, resp.) is satisfied.

Let x_0 and y_0 be the outcomes of the uncertain objects x and y for which $\text{dist}(x_0, y_0) = maxdist(x, y)$ holds. Assume that $maxdist(x, y) > maxdist(x, z) + maxdist(z, y)$. Given an arbitrary outcome z_0 of z, since dist is a metric by assumption, by the triangle inequality it holds that $\text{dist}(x_0, z_0) + \text{dist}(z_0, y_0) \geq \text{dist}(x_0, y_0) = maxdist(x, y)$, and, by the above assumption, it finally holds that $\text{dist}(x_0, z_0) + \text{dist}(z_0, y_0) > \text{dist}(x_1, z_1) + \text{dist}(z_2, y_2)$. But, this would contradict the definition of x_1, z_1, z_2 and y_2, since, by definition of $maxdist$, it is the case that $\text{dist}(x_1, z_1) \geq \text{dist}(x_0, z_0)$ and $\text{dist}(z_2, y_2) \geq \text{dist}(z_0, y_0)$ and, hence, that $\text{dist}(x_1, z_1) + \text{dist}(z_2, y_2) \geq \text{dist}(x_0, z_0) + \text{dist}(z_0, y_0)$. Hence, the statement follows. □

Thus, even if the space obtained by using $maxdist$ as a distance function is not Euclidean, it is anyway a metric one provided that dist is itself a metric (as it is the case when dist is the Euclidean distance). The above result has the important practical implication that the set $OutCands$ can be determined by exploiting certain distance-based outlier detection algorithms designed to work in general metric spaces.

As a consequence, in step 2 the algorithm UDBOD employs the DOLPHIN technique (Angiulli and Fassetti 2009). DOLPHIN performs two sequential scans of the dataset. During the first scan, a superset of the true outliers is detected. by accumulating in a data structure, called INDEX, the incoming objects that cannot be recognized as outliers by exploiting the objects already stored in INDEX. The second scan is needed to recognize the true outliers in INDEX. The temporal cost is derived by proving that the size of INDEX is $O(\frac{k}{p})$.

Algorithm 3: *Candidate Filtering Phase.*

```
1  Set Outliers to the empty set
2  foreach x in OutCands do
3      if D_k^0(x) > R then
4          Insert x into Outliers;
5      else
6          outlier = true;
7          Generate m outcomes w_1, ..., w_m of x
8          Initialize matrix P (having m rows and k cols)
9          Let DS_{x,R} be {y_1, ..., y_ℓ}
10         j = 1
11         while j ≤ ℓ and outlier do
12             LB = 0;
13             for h = 1 to m do
14                 Let p = Pr(d(w_h, y_j) ≤ R)
15                 lb = 0
16                 for i = min(j, k − 1) downto 1 do
17                     P[h, i] = p · P[h, i − 1] + (1 − p) · P[h, i]
18                     lb = lb + P[h, i]
19                 P[h, 0] = P[h, 0] · (1 − p)
20                 lb = lb + P[h, 0]
21                 LB = LB + (1 − lb)
22             if j ≥ k and LB > m · (1 − δ) then
23                 outlier = false
24             j = j + 1
25         if outlier then
26             Outliers = Outliers ∪ {x}
27 return the set Outliers
```

5.3 Candidate filtering phase

The *Candidate filtering phase* (see steps 3-9 in Algorithm 1) computes the set $Outliers$ of uncertain outliers contained in the dataset by processing the objects in the set $OutCands$. In order to reduce the computational effort, a lower bound property is introduced and exploited, which avoids to consider all the potential neighbors of the candidate outliers in order to compute their outlier probability.

The objects x of $OutCands$ such that $D_k^0(x) > R$ can be safely inserted into $Outliers$ since, as stated in Section 2, they are outliers for sure. We call these objects *ready outliers*.

As for the *non-ready outliers* x, it has to be decided whether $Pr(D_k(x) \leq R) \leq 1 - \delta$ or not, and this is accomplished by computing (4) exploiting the procedure explained in the following of this section and reported in Algorithm 3 (see lines 6-26). With this aim, consider the set $\mathbf{DS}_{x,R} = \{y \in \mathbf{DS} \mid mindist(x, y) \leq R\}$, also called the *neighbor list* of x (in $\mathbf{DS}$ w.r.t. R).

The objects in the set $\mathbf{DS}_{x,R}$ are all and only the uncertain objects of $\mathbf{DS}$ which give a contribution to the probability $Pr(D_k(x) \leq R)$, since for the objects $z \in \mathbf{DS} \setminus \mathbf{DS}_{x,R}$ it holds that $Pr(\text{dist}(x, z) \leq R) = 0$.

Let $w_1, \ldots, w_m$ denote m outcomes of x, and let $y_1, \ldots, y_\ell$ denote the uncertain objects in the set $\mathbf{DS}_{x,R}$ ordered accordingly to an arbitrary criterion.

Let $P(w_h, i, j)$ denote the probability that the certain object w_h has exactly i neighbors among the first j uncertain objects $y_1, \ldots, y_j$ of $\mathbf{DS}_{x,R}$

Moreover, let $P_k^j(x)$ denote the probability that x has at least k neighbors within distance R among the first j uncertain objects of $\mathbf{DS}_{x,R}$, then by exploiting the approximation in (5):

$$P_k^j(x) = \frac{1}{m} \sum_{h=1}^{m} \left(1 - \sum_{i=0}^{k-1} P(w_h, i, j) \right) \tag{9}$$

The following theorem holds.

Theorem 4 *If there exists $j \leq k$ such that $P_k^j(x) > 1 - \delta$ then x is not an outlier.*

Proof The proof follows by noticing that, for each $j \in \{1, 2, \ldots, |\mathbf{DS}_{x,R}|\}$, it holds that $P_k^j(x) \leq Pr(D_k(x) \leq R)$, that is to say that $P_k^j(x)$ is a lower bound for the probability that x has exactly i neighbors in a generic outcome of $\mathbf{DS}$. □

Consequently, if for some $j \leq k$ the left hand side term above exceeds $1 - \delta$, then the computation can be early stopped reporting that x is not an outlier.

Notice that $P_k^\ell(x)$ is precisely $Pr(D_k(x) \leq R)$. Interestingly, in order to compute $P_k^\ell(x)$ and its lower bounds $P_k^j(x)$ $(1 \leq j \leq \ell)$ only space $O(mk)$ is needed instead of $O(mk\ell)$, since the $mk\ell$ terms $P(w_h, i, j)$ can be computed by means of the incremental procedure described next. Let p_j be $Pr(\text{dist}(w_h, y_j) \leq R)$, then the following relationship is satisfied:

$$P(w_h, i, j) = p_j \cdot P(w_h, i - 1, j - 1) + (1 - p_j) \cdot P(w_h, i, j - 1),$$

that is to say, the probability that the certain object w_h has exactly i neighbors among the first j uncertain objects $y_1, \ldots, y_j$ is equal to (i) the probability p_j that y_j is a neighbor of w_h and w_h has exactly $i - 1$ neighbors among the uncertain objects $y_1, \ldots, y_{j-1}$, plus (ii) the probability $1 - p_j$ that y_j is not a neighbor of w_h and w_h has exactly i neighbors among the uncertain objects $y_1, \ldots, y_{j-1}$. By the above relationship it is clear that in order

to compute the terms $P(w_h, \cdot, j)$ only the terms $P(w_h, \cdot, j-1)$ are needed, that are k terms for each of the m outcomes w_h of x.

The *Candidate Filtering Phase*, reported in Algorithm 3, details the procedure to compute the lower bound $P_k^j(x)$ (there, the variable LB is used to accumulate the value of the lower bound, while the matrix elements $P[h, i]$ to store the values $P(w_h, i, \cdot)$).

The above procedure does not depend on the order $y_1, \ldots, y_\ell$ of the objects in $\mathbf{DS}_{x,R}$, but considering first the objects closest to x may help to accelerate convergence of the lower bound. With this aim, uncertain objects in the set $\mathbf{DS}_{x,R}$ are sorted in ascending order of their score $s(y_j)$ defined as:

$$s(y_j) = \frac{maxdist(x, y_j) - R}{maxdist(x, y_j) - mindist(x, y_j)},$$

for $maxdist(x, y_j) > R$, and $s(y_j) = 0$ for $maxdist(x, y_j) \leq R$. The score $s(y_j)$ ranges in [0, 1].

5.4 Temporal cost

Let d denote the cost of computing the distance dist between two certain objects of $\mathbb{D}$ and also the distances $maxdist$ and $mindist$ between two uncertain objects of $\mathbb{D}$. Let c denote the number of outlier candidates, let m denote the number of samples employed to evaluate integrals by means of the formula in (6), and let $\overline{\ell}$ denote the mean number of elements in the neighbor lists $\mathbf{DS}_{x,R}$ employed to compute the outlier probability, for x a candidate outlier.

The parameter selection phase costs $O(s^2 d)$, where $s \ll N$ is the size of the sample employed to estimate R^*, size that can be considered fixed. The candidate selection phase costs $O(\frac{k}{p} N d)$, where $p \in (0, 1]$ is an intrinsic parameter of the dataset at hand (Angiulli and Fassetti 2009). As for the candidate filtering phase, for each outcome w_h of x ($1 \leq h \leq m$) and for each $y_j \in \mathbf{DS}_{x,R}$ ($1 \leq j \leq \overline{\ell}$), computing $Pr(\text{dist}(w_h, y_j) \leq R)$, with $y_j \in \mathbf{DS}_v$, costs $O(md)$, while obtaining the terms $P(w_h, \cdot, j)$ costs $O(k)$. Thus, deciding for $Pr(D_k(x) \leq R) \leq 1 - \delta$ costs in the worst case $O(\overline{\ell} m(md + k))$. As a whole, the candidate filtering phase costs $O(c\overline{\ell} m(md + k))$.

Thus, the cost of the algorithm is $O\left(s^2 d + \frac{k}{p} N d + c\overline{\ell} m(md + k)\right)$. The last phase of the algorithm is the potentially heaviest one, since it involves integral calculations. To be practical, the algorithm must be able to select a number of outlier candidates c close to the value αN of expected outliers ($\alpha \in [0, 1]$) and possibly to keep as lower as possible the value of $\overline{\ell}$.

6 Experimental results

In this section, we describe experimental results carried out by using the UDBOD algorithm. If not otherwise stated, we use the default values for parameters in Table 1 and $m = 1{,}000$. The experiments are conducted on a Intel Xeon 2.33 GHz based machine with 4GB of RAM under the GNU/Linux operating system. Each dataset is characterized by a parameter γ, called *spread*, used to set the degree of uncertainty associated with dataset objects.

Experiments are organized as follows. Section 6.1 studies the scalability of the method. Section 6.2 studies how parameters influence the number of candidate and ready outliers. Section 6.3 compares the proposed method with related literature. Finally, Section 6.4 presents two cases of study.

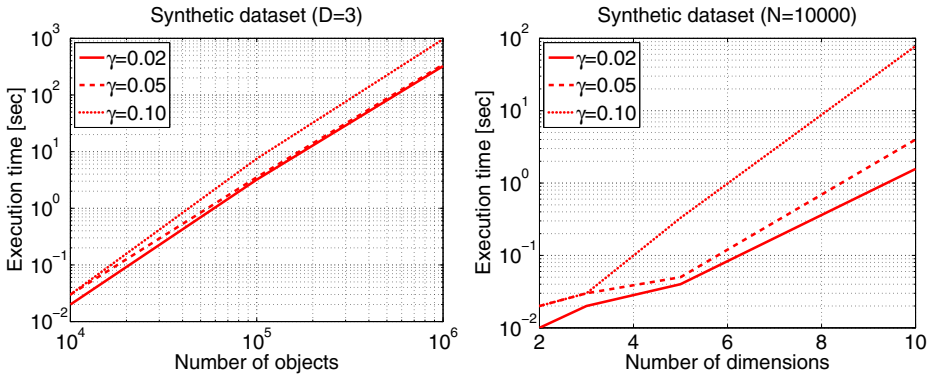

Fig. 1 Scalability with respect to the dataset size and the number of dimensions for the *Synthetic* dataset

6.1 Scalability analysis

We considered a family of synthetic data whose elements differ for the number N of uncertain objects and the number D of attributes, generated according to the following strategy. The uncertain objects in each dataset form two normally distributed separated clusters with mean $(-10, 0, \ldots, 0)$ and $(10, 0, \ldots, 0)$, respectively. Moreover, the 3‰ of the dataset objects are uniformly distributed in a region lying on the hyper-plane $x = 0$ (that is to say, their first coordinate is always zero). Uncertain objects are randomly generated and may use a normal, an exponential or a uniform distribution whose spread is related to the standard deviation of the overall data by means of the parameter $\gamma \in \{0.02, 0.05, 0.1\}$.

Figure 1 on the left shows the scalability with respect to the number N of objects. In this experiment, N has been varied between 10,000 and 1,000,000, while the number of dimensions D has been held fixed to 3. These curves show that the method has very good performances for different values of spread. In particular, the execution time is below 1,000 seconds even for one million of objects, confirming that the method is able to manage large datasets.

Figure 1 on the right shows the scalability with respect to the number of dimensions D. This time the number of objects has been held fixed to 10,000. Also in this case, time performances are good. The execution time clearly increases with the dimensionality, due to the increasing cost of evaluating outcomes of the distributions, but in these experiments it remained below 100 seconds even for 10-dimensional datasets.

Fig. 2 Accuracy of the *Synthetic* dataset family

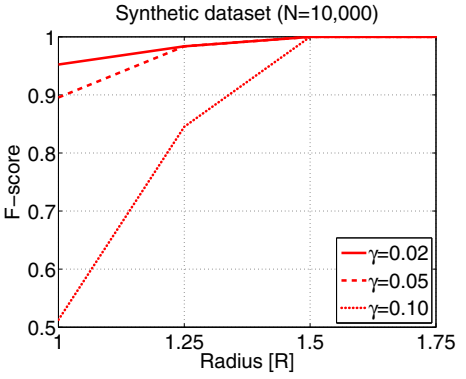

Table 2 Outliers detected for the *Synthetic* dataset

Spread \ Radius	1.0	1.25	1.5	1.75
0.02	3.3‰	3.1‰	3.0‰	3.0‰
0.05	3.7‰	3.1‰	3.0‰	3.0‰
0.1	8.7‰	4.1‰	3.0‰	3.0‰

We studied also the accuracy. Figure 2 reports the F-score as a function of the radius R. It is assumed that the outliers are the objects lying on the hyperplane $x = 0$. The curves highlight the accuracy of the approach. Indeed, for values of radius above 1.5 the F-score is close to 1 for every considered spread, and for spread equal to 0.02 and 0.05 the F-score is almost always above 0.9 for every radii considered.

For the highest spread and the lowest radius considered, the F-score lowers. This situation can be understood by considering Table 2 which reports the number of outliers returned by the method. It can be seen that for spread equal to 0.1 and radius set to 1.0, the number of outliers returned by the method is notably larger than the actual number of outliers. All the objects lying on the hyperplane $x = 0$ are correctly retrieved, but the method start to consider as outliers the objects lying in the tails of the distributions associated with the clusters.

6.2 Sensitivity analysis

In this section, we study how parameters influence performances, that is the number of candidate outliers and of ready outliers. We employed the following datasets: *Cities* ($N = 5{,}922$, $d = 2$), *Household* ($N = 2{,}075{,}259$, $d = 7$) *Skin* ($N = 245{,}057$, $d = 3$), and *US Points* ($N = 15{,}206$, $d = 2$). *Cities*, containing 5,922 city and village locations in Greece, and *US Point*, containing 15,206 points of populated places in USA, are from the R-Tree Portal.[1] *Household* and *Skin*, are from the UCI ML Repository.[2] *Household* contains 2,075,259 measurements of electric power consumption. *Skin* is collected by randomly sampling 245,057 RGB values from face images of various age groups, race groups, and genders.

For all datasets, a family of uncertain datasets has been obtained as follows. An uncertain object x_i has been associated with each certain object v_i in the original dataset, whose pdf $f^{x_i}(u)$ is a multidimensional normal, uniform or exponential randomly selected distribution centered in x_i and whose spread is related to the standard deviation of the overall data by means of the parameter γ. Different values for the parameter β and for the spread γ (specifically, $\gamma \in \{0.05, 0.1\}$) have been taken into account.

Figure 3 reports the number of candidate outliers detected at the end of the candidate selection phase (gray bar, on the left), the actual number of outliers detected (green bar, on the middle), and the number of non-ready candidates (red bar, on the right). Specifically, the non-ready candidates are the objects for which (4) has to be evaluated. Notice that in almost all of the runs the number of candidate objects represents a small fraction of the overall dataset size, in the worst case amounting to the 0.65% (when $\gamma = 0.05$) and the 1.67% (when $\gamma = 0.10$) for *Cities*, the 0.32% for *Household*, the 0.38% (when $\gamma = 0.05$) and the

[1] See http://www.rtreeportal.org.

[2] See http://archive.ics.uci.edu/ml.

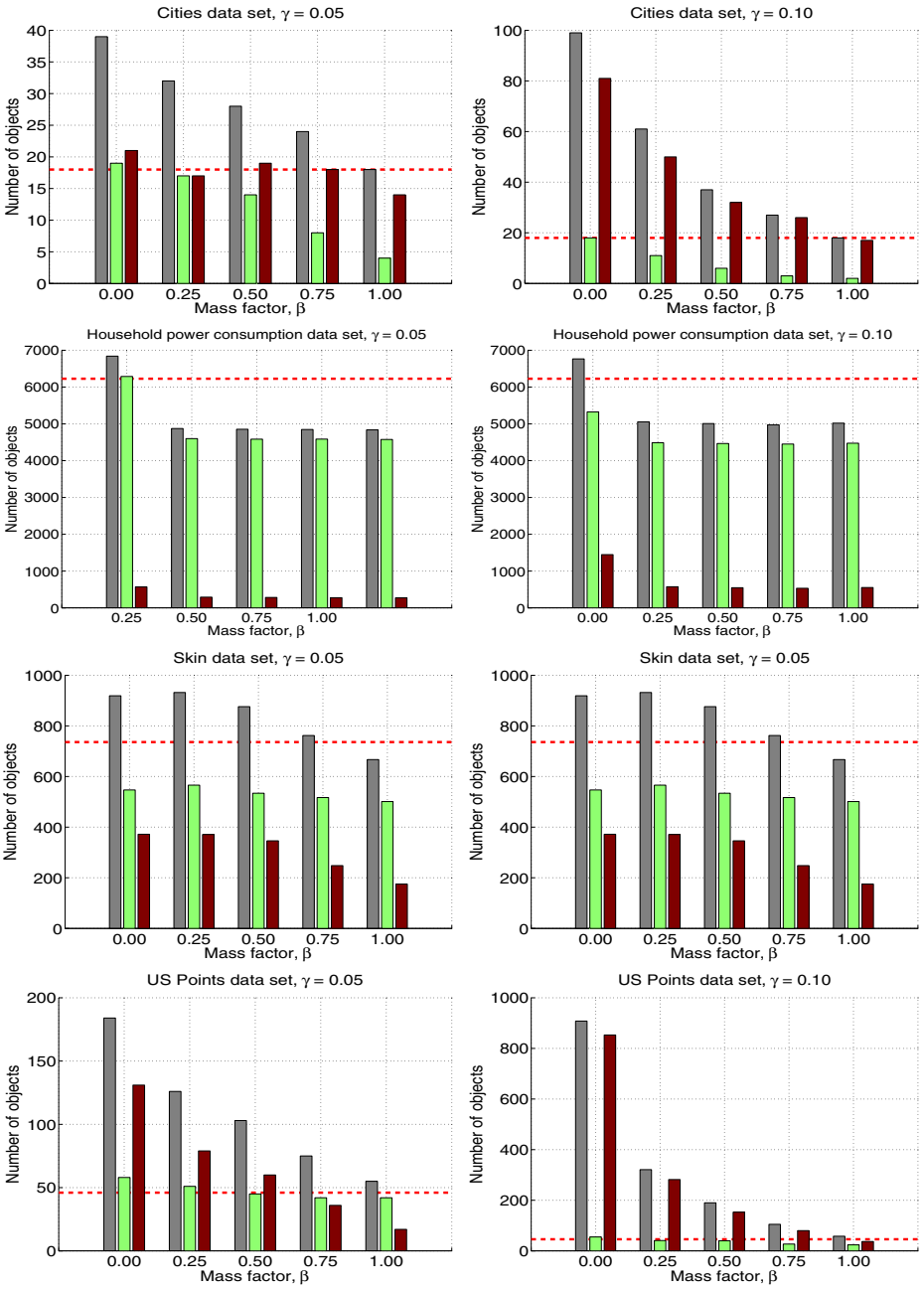

Fig. 3 Number of candidates (gray bars, on the left), outliers (green bars, on the middle), and non-ready candidates (red bars, on the right). The dashed line represents the number αN of expected outliers

0.45% (when $\gamma = 0.10$) for *Cities*, and the 1.21% (when $\gamma = 0.05$) and the 5.97% (when $\gamma = 0.10$) for *Cities*. This confirms that the candidate selection phase allows to save a vast amount of time.

The dashed line represents the number αN, with $\alpha = 3‰$. From the figure it is clear the effect of the parameter β on the efficiency of the method (number of candidates) and on the number of actual outliers. It appears there is a trade-off between these two numbers that can be controlled by means of β. As far as the correspondence between the number of actual outliers and the number αN of expected ones, according to Theorem 1 the number of outliers for $\beta = 0$ ($\beta = 1$, resp.) should be greater (lower, resp.) than the expected αN. Clearly, this is true modulo (i) the error introduced by the radius estimation and (ii) the introduction of the correction to the estimation. Specifically, the above relationship is satisfied for *Cities*, *US Points* and *Household* with $\gamma = 0.05$, and for *Cities* and *US Points* with $\gamma = 0.10$. As for the other cases, the number of actual outliers is always smaller than the expected one, since the correction of the estimation has been employed. This is also confirmed by the fact that the number of actual outliers is almost the same for the different values of β. Thus, in these cases there are less than αN true outliers and the parameter estimation phase is able to determine the right radius. Thus, the above experiment highlights that the parameter estimation phase allows to determine values for the parameters complying with the required number of outliers without exceeding the number of clearly non-outlying objects.

As for the number of candidates, it is about inversely proportional to the value of β. So, in order to reduce the computational effort it is better to employ β values greater than zero. As for values of β close to one, the actual number of outliers could result sensibly smaller than the αN fraction, so it is better to employ β values smaller than one. Intermediate values for β (around 0.5) seem a good trade-off between the number of candidates and the number of actual outliers. Indeed, $\beta = 0$ could result in a lot of candidate outliers (e.g., for *US Points* and $\gamma = 0.1$ the number of candidates is more than 16 times greater than the number of outliers), while $\beta = 1$ could result in too few outliers (e.g., for *Cities* and $\gamma = 0.1$ the number of outliers is about nine times smaller than the expected one).

Figure 4 shows the size of the neighbor list associated with rejected candidates (green bar, on the left), namely the non-ready candidates which are inliers, and number of neighbors considered until early stop is reached (red bar, on the right). The dashed line represents the value of the parameter k. The figures show that the candidate filtering phase is able to recognize the inliers without the need to take into account all the ℓ objects in the neighbor list (whose average number corresponds to the blue bars in Fig. 4). In particular, how witnessed by red bars (on the right), the number of neighbors actually considered in (9) is close to k. Notice that at least k neighbors have to be considered in order to prove the inlierness of an object. Thus the candidate filtering phase allows to maintain very low the computational effort to be paid on candidate objects.

Figure 5 shows the elapsed time at the end of the parameter estimation phase (dotted line), the candidate selection phase (dashed line), and the candidate filtering phase (solid line). The plots confirm that the bulk of the computation is given by the last phase.

6.3 Comparison with other methods

We compared UDBOD with the *DensitySamp* technique introduced in Aggarwal and Yu (2008) and the *Determistic* technique introduced in Aggarwal and Yu (2001b). The technique (Aggarwal and Yu 2008) is designed for uncertain data and described in Section 3. The technique (Aggarwal and Yu 2001b) does not manage uncertainty, but determines outliers by finding projections of the data which have abnormally low density, and was already used as a baseline competitor in Aggarwal and Yu (2008). *Deterministic* determines outliers by finding projections of the data which have abnormally low density. In the comparison we

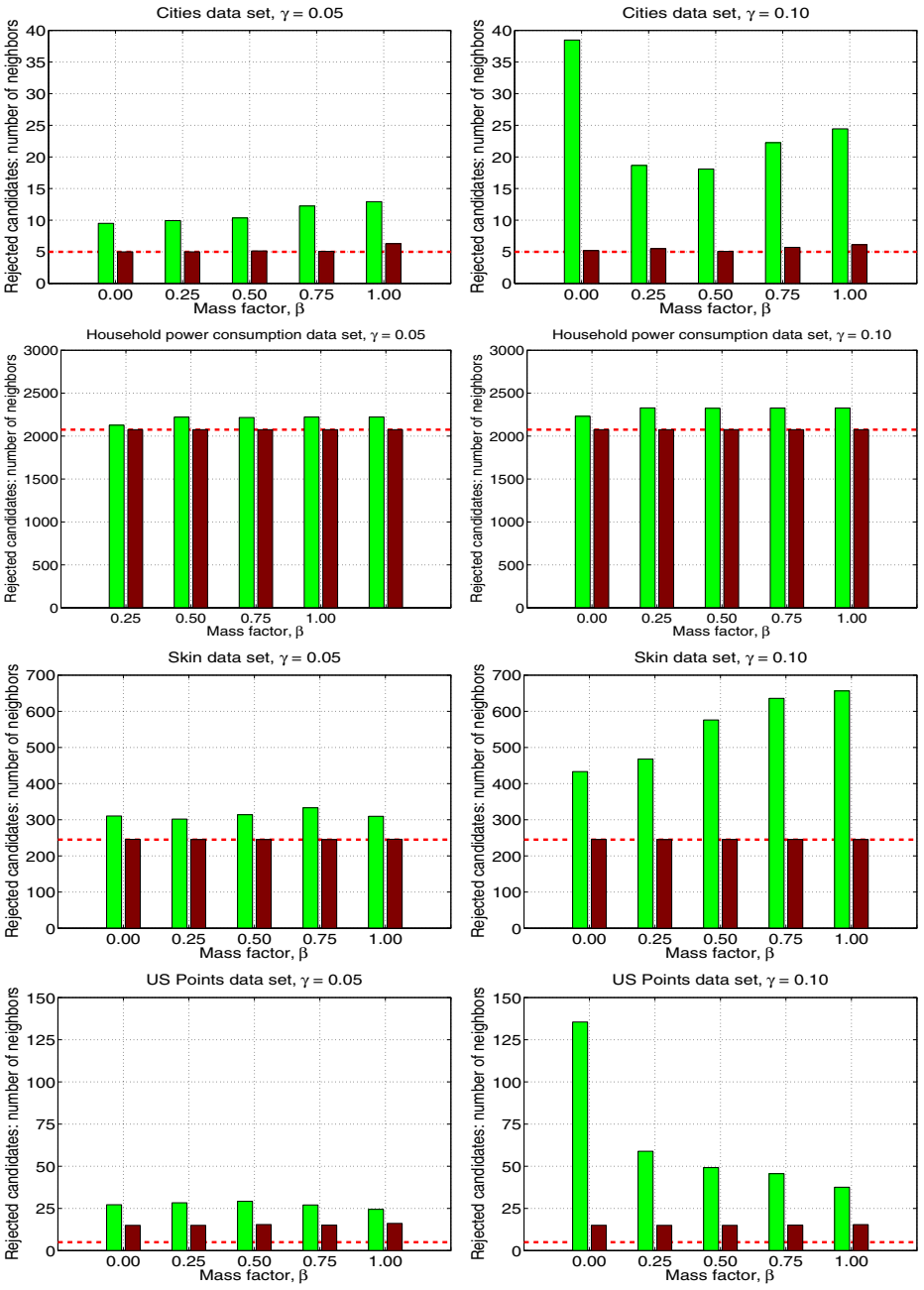

Fig. 4 Rejected candidates: size of the neighbor list (green bars, on the left) and number of neighbors considered until early stop (red bars, on the right). The dashed line represents the value of the parameter k

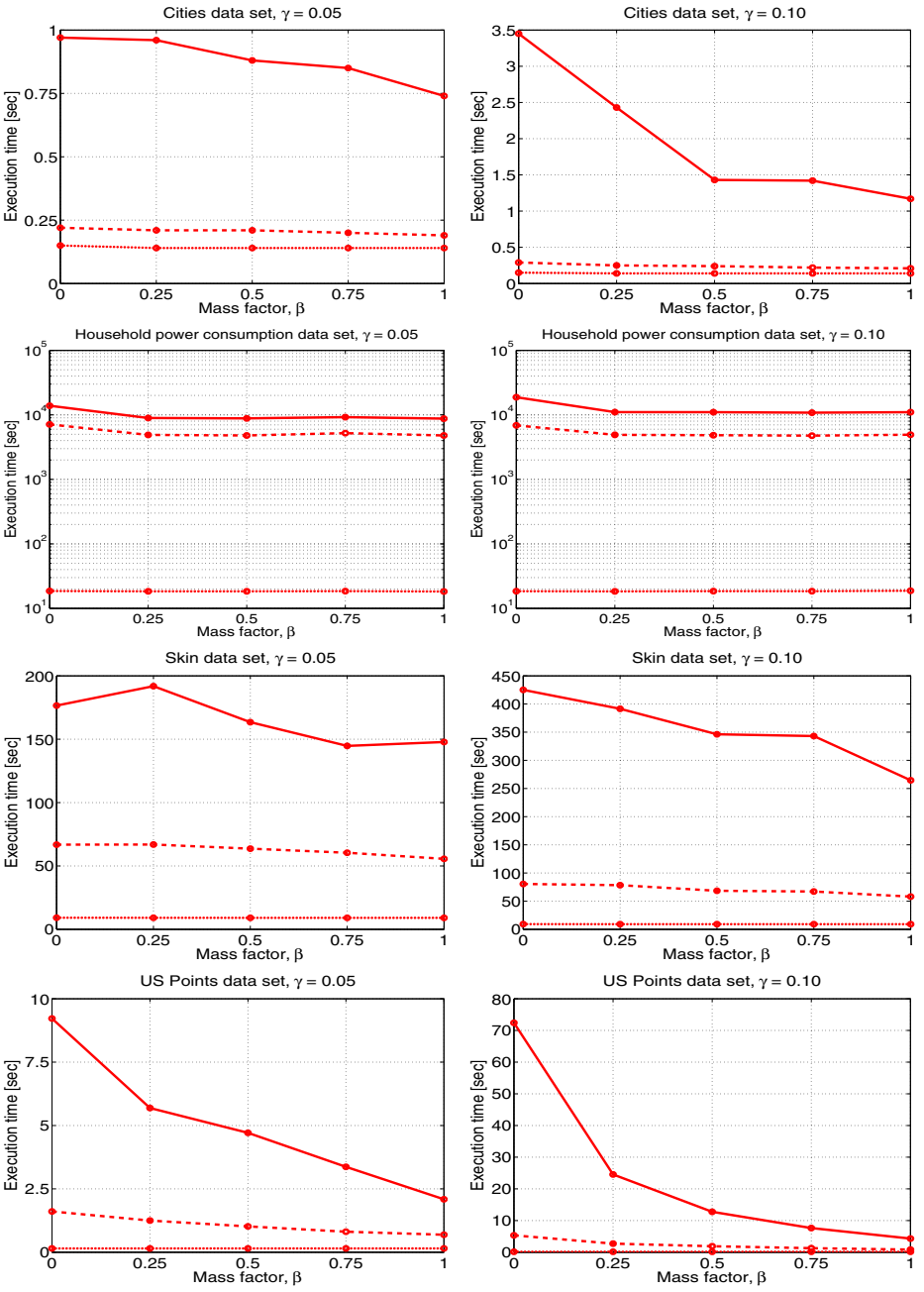

Fig. 5 Execution time: elapsed time at the end of the parameter estimation phase (dotted line), at the end of the candidate selection phase (dashed line), and at the end of the candidate filtering phase (solid line)

employed a family of datasets described in Aggarwal and Yu (2008), whose characteristics are recalled next. The data points were generated by creating Gaussian clusters in the underlying data. whose centers were generated uniformly in the unit data cube. The number of data points in each cluster was proportional to a random variable drawn from a uniform distribution in [0, 1]. The radius along each dimension was drawn from a uniform distribution in $[0, r]$. A fraction p of the data points were designated as outliers. The outliers were generated anywhere in the data cube. A total of N data points were generated in d dimensions. All datasets were normalized, so that the standard deviation along each dimension was 1 unit. Each uncertain attribute is normally distributed with zero mean and standard deviation drawn from a uniform distribution in $[0, 2 \cdot f] \cdot \sigma$, where σ is the standard deviation of that dimension in the underlying data. The dataset is denoted by $R(r).O(p).d(d).D(N).U(f)$.

Since the outliers were known, the precision and recall could be measured. In the case of UDBOD, the trade-off between the precision and recall is measured by varying the radius R. As for the two other algorithms, we varied their parameters and applied *Deterministic* to the above datasets as described in Aggarwal and Yu (2008). Figure 6 reports the results of the comparison. According to Aggarwal and Yu (2008), we employed the following values for the parameters: $r = 0.3$, $d = 10$, $p \in \{0.1, 0.2\}$, $N = 100$K, and $f \in \{1.0, 1.5, 2.0, 2.5, 3.0\}$. Specifically, the two plots on the top report the Precision and the Recall of the methods for different outlier fractions, namely $p = 0.1$ and $p = 0.2$, and uncertainty level $f = 1.5$. As for the two plots on the bottom, the F-score obtained by the methods for the same outlier fractions p and uncertainty levels f ranging in [1.0, 3.0].

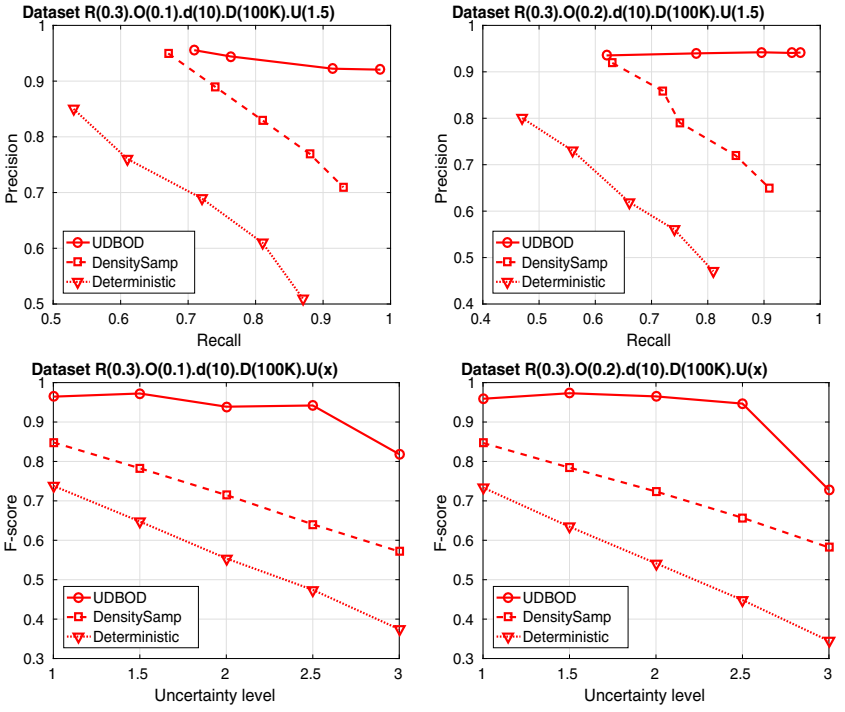

Fig. 6 Comparison with *DensitySamp* and *Deterministic*

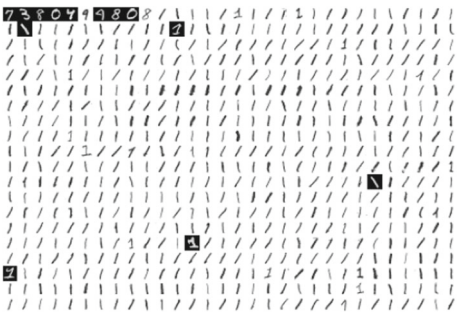

Fig. 7 *Uncertain MNIST* and detected outliers

6.4 Cases of study

Handwritten digits. MNIST is an high-dimensional dataset of handwritten digits represented as images of 28×28 pixels extensively employed in the literature[3]. We simulated an uncertain scenario in which digits are blurred, by associating a normally distributed uncertain object o_i with mean μ_i and standard deviation σ_i to each non-overlapping 2×2 tile of image pixels. The parameters μ_i and σ_i are obtained as the mean and the standard deviation of the intensities of the pixels within the corresponding tile. Thus, the dataset consists of 196-dimensional uncertain objects. We randomly selected 590 digits from the class "1" and 10 digits from the remaining classes to form a dataset of 600 uncertain objects.

Figure 7 shows the dataset objects (pixels intensities are those corresponding to μ_i values). Digits corresponding to the outliers have been highlighted by complementing their intensity values (so that they appear on a dark background). Outliers are computed for $k = 5$ and for the radius value determined by the algorithm with $\alpha = 0.02$ (corresponding to about 12 objects) and $\beta = 0.5$. It can be seen that eight out of the ten non-"1" digits have been detected. The only exception is represented by a "8" digit with markedly uncertain borders and a largely distorted "9" digit. As for remaining outliers, they correspond to "1" digits that are not usual within the collection.

The number of candidates returned by the candidate selection phase was 110. This number witnesses the difficulty of the problem, since it follows from the fact that the support of the objects are largely overlapping. Despite this number, during candidate filtering the mean number of neighbors considered until early stop was only 7.4. By using $m = 100$, the execution time of UDBOD was about 104.4 seconds (11.7 secs for parameter estimation, 5 secs for candidate selection, and 87.7 secs for candidate filtering).

Mobile ad-hoc network data. A Mobile Ad hoc NETworks (MANET) (Bai and Helmy 2006) is a collection of wireless mobile nodes forming a self-configuring network. Applications include mobile classrooms, battlefield communication, disaster relief, and others. The *mobility model* of a MANET is designed to describe the movement pattern of mobile users, and how their location, velocity and acceleration change over time. A popular mobility model is the *Random Waypoint* model (Bettstetter et al. 2004), in which nodes move independently within a certain area, called *support area*. For a squared support area of size

[3] http://yann.lecun.com/exdb/mnist.

a by a, with a its *diameter*, centered in (x_0, y_0), the pdf of the random waypoint model is provided by the following analytical expression:

$$f_{\mathrm{rw}}(x, y) \approx \frac{36}{a^6} \cdot \left((x - x_0)^2 - \frac{a^2}{4}\right) \cdot \left((y - y_0)^2 - \frac{a^2}{4}\right),$$

for $x \in \left[x_0 - \frac{a}{2}, x_0 + \frac{a}{2}\right]$ and $y \in \left[y_0 - \frac{a}{2}, y_0 + \frac{a}{2}\right]$, and $f_{\mathrm{rw}}(x, y) = 0$ outside.

The nodes of a MANET are typically distinguished by their limited power, processing, and memory resources. Multiple hops are usually needed for a node to exchange information with any other node and nodes take advantage of their neighbors in order to communicate with the rest of the network. A node can correctly receive packets if the signal strength of the packet at that node is above a certain threshold and the needed transmission power is inversely proportional to the squared distance separating the transmitter to the receiver.

The dataset (see Fig. 8a) consists of 250 MANET nodes distributed along three different paths joining two locations. Each red square in the figure delimits the support area associated with a node (diameters of support areas range from the 2% to 6% of the simulation area side). Since, information exchange is accomplished by multiple hops involving neighbor nodes, the smaller the number of neighbors lying in the neighborhood of a node, the less reliable, in terms of QoS (Quality of Service), the region which the node belongs to. Thus, we exploit uncertain distance-based outlier detection to determine the less reliable regions of the simulation area. With this aim, we fixed the radius R to the 7% of the simulation area side (a circular region of radius R is highlighted in 8a), a value corresponding to a predefined level of transmission power due to device constraints.

Since, the QoS can be related to the number of neighbors, we detected the uncertain distance-based outliers for increasing values of k. Figures 8c and d show the outliers for $k = 3$ and $k = 10$, respectively. The outliers for $k = 3$ are positioned along the central path, which corresponds to the lowest populated region of the area, while the additional outliers for $k = 10$ are located along the path on the right, which corresponds to the mild populated region of the area. As for remaining objects, they are located along the path on the left, which corresponds to the most reliable route between the two extrema. As for Fig. 8b, it provides a picture of the QoS associated with each location of the area, since the color of each point (colors range from blue, for $k = 1$, to red, for $k = 35$) is proportional to the smallest value of k for which the location, regarded as an uncertain object, becomes an outlier.

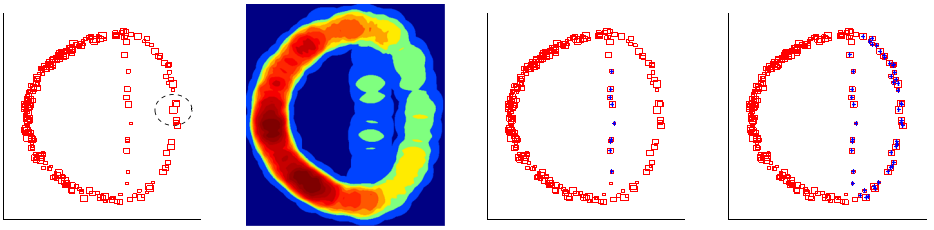

Fig. 8 MANET dataset: **a** nodes distributed along 3 paths; **b** QoS associated with locations (colors range from red, for higher QoS values, to blue); uncertain outliers (blue asterisks) for $k = 3$ **c** and $k = 10$ **d**

7 Conclusions

A novel definition of uncertain outlier has been introduced dealing with multidimensional arbitrary shaped pdfs and representing the generalization of the classic distance-based outlier definition. Our approach corresponds to perform a nearest neighbor density estimate on all the possible outcomes of the dataset and, to the best of our knowledge, has no counterpart in the literature. Possible future research directions include techniques for alleviating the cost involved with the computation of integrals, possibly based on exploiting data indexing techniques, and of alternative notions of uncertain outlier, as ones inspired to adaptive density estimation strategies, or by considering more involving scenarios including time-varying distributions.

References

Aggarwal, C.C. (2014). Data clustering: algorithms and applications. Chapman & Hall/CRC, Ch. A Survey of Uncertain Data Clustering Algorithms.

Aggarwal, C.C. (2016). *Outlier analysis*, 2nd edn. New York: Springer Publishing Company, Incorporated.

Aggarwal, C.C., & Yu, P. (2001). Outlier detection for high dimensional data. In *SIGMOD*.

Aggarwal, C.C., & Yu, P.S. (2001). Outlier detection for high dimensional data. In *Proceedings of the ACM SIGMOD international conference on management of data* (pp. 37–46).

Aggarwal, C., & Yu, P. (2008). Outlier detection with uncertain data. In *SDM* (pp. 483–493).

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

Angiulli, F. (2020). CFOF: a concentration free measure for anomaly detection. *ACM Transactions on Knowledge Discovery from Data*, *14*, 4:1–4:53.

Angiulli, F., Basta, S., Pizzuti, C. (2006). Distance-based detection and prediction of outliers. *IEEE Transactions on Knowledge and Data Engineering*, *18*(2), 145–160.

Angiulli, F., & Fassetti, F. (2007). Nearest neighbor-based classification of uncertain data, ACM Transactions on Knowledge Discovery from Data 7 (1).

Angiulli, F., & Fassetti, F. (2009). Dolphin: an efficient algorithm for mining distance-based outliers in very large datasets. ACM Transactions on Knowledge Discovery from Data 3(1), Article 4.

Angiulli, F., & Fassetti, F. (2012). Indexing uncertain data in general metric spaces. *IEEE Transactions on Knowledge and Data Engineering*, *24*(9), 1640–1657.

Angiulli, F., & Fassetti, F. (2013). Outlier detection with arbitrary probability functions. In *AI*IA* (pp. 421–432).

Angiulli, F., & Fassetti, F. (2014). Exploiting domain knowledge to detect outliers. *Data Mining and Knowledge Discovery*, *28*(2), 519–568.

Angiulli, F., Fassetti, F., Palopoli, L. (2009). Detecting outlying properties of exceptional objects. ACM Transactions on Database Systems 34 (1).

Angiulli, F., Fassetti, F., Palopoli, L. (2013). Discovering characterizations of the behavior of anomalous subpopulations. *IEEE Transactions on Knowledge and Data Engineering*, *25*(6), 1280–1292.

Angiulli, F., & Pizzuti, C. (2005). Outlier mining in large high-dimensional data sets. *IEEE Transactions on Knowledge and Data Engineering*, *2*(17), 203–215.

Arning, A., Aggarwal, C., Raghavan, P. (1996). A linear method for deviation detection in large databases. In *KDD* (pp. 164–169).

Bai, F., & Helmy, A. (2006). *Wireless ad hoc and sensor networks*. New York: Springer. Ch. a survey of mobility modeling and analysis in wireless adhoc networks.
Barnett, V., & Lewis, T. (1994). Outliers in statistical data. New York: Wiley.
Bay, S.D., & Schwabacher, M. (2003). Mining distance-based outliers in near linear time with randomization and a simple pruning rule. In *KDD*.
Bettstetter, C., Hartenstein, H., Pérez-Costa, X. (2004). Stochastic properties of the random waypoint mobility model. *Wireless Networks*, *10*(5), 555–567.
Bi, J., & Zhang, T. (2004). Support vector classification with input data uncertainty. In *NIPS* (pp. 161–168).
Breunig, M.M., Kriegel, H., Ng, R., Sander, J. (2000). Lof: identifying density-based local outliers. In *SIGMOD*.
Chandola, V., Banerjee, A., Kumar, V. (2009). Anomaly detection: a survey, ACM Computing Surveys 41 (3).
Davies, L., & Gather, U. (1993). The identification of multiple outliers. *Journal of the American Statistical Association*, *88*, 782–792.
Ghoting, A., Parthasarathy, S., Otey, M. (2006). Fast mining of distance-based outliers in high-dimensional datasets. In *SDM, Bethesda, MD, USA*.
Green, T., & Tannen, V. (2006). Models for incomplete and probabilistic information. *IEEE Data Engineering Bulletin*, *29*(1), 17–24.
Hawkins, D. (1980). *Identification of outliers. monographs on applied probability and statistics*. London: Chapman & Hall.
Hawkins, S., He, H., Williams, G.J., Baxter, R.A. (2002). Outlier detection using replicator neural networks. In *Proceedings of the 4th international conference on data warehousing and knowledge discovery* (pp. 170–180).
Jiang, B., & Pei, J. (2011). Outlier detection on uncertain data: objects, instances, and inference. In *ICDE*.
Khan, A., Ye, Y., Chen, L. (2018). On uncertain graphs. synthesis lectures on data management. Morgan & Claypool.
Knorr, E., & Ng, R. (1999). Finding intensional knowledge of distance-based outliers. In *VLDB* (pp. 211–222).
Knorr, E., Ng, R., Tucakov, V. (2000). Distance-based outlier: algorithms and applications. *VLDB Journal*, *8*(3-4), 237–253.
Kriegel, H.-P., & Pfeifle, M. (2005). Density-based clustering of uncertain data. In *KDD* (pp. 672–677).
Lepage, G. (1978). A new algorithm for adaptive multidimensional integration. Journal of Computational Physics 27.
Lindley, D. (2006). *Understanding uncertainty*. New York: Wiley-Interscience.
Liu, F., Ting, K., Zhou, Z.-H. (2012). Isolation-based anomaly detection. ACM Transactions on Knowledge Discovery from Data (TKDD) 6 (1).
Liu, B., Xiao, Y., Cao, L., Hao, Z., Deng, F. (2013). Svdd-based outlier detection on uncertain data. *Knowledge and Information Systems*, *34*(3), 597–618.
Mohri, M. (2003). Learning from uncertain data. In *COLT* (pp. 656–670).
Papadimitriou, S., Kitagawa, H., Gibbons, P., Faloutsos, C. (2003). Loci: fast outlier detection using the local correlation integral. In *ICDE* (pp. 315–326).
Ramaswamy, S., Rastogi, R., Shim, K. (2000). Efficient algorithms for mining outliers from large data sets. In *SIGMOD* (pp. 427–438).
Tao, Y., Xiao, X., Zhou, S. (2006). Mining distance-based outliers from large databases in any metric space. In *KDD Philadelphia, PA, USA* (pp. 394–403).
Tax, D.M.J., & Duin, R.P.W. (2004). Support vector data description. *Machine Learning*, *54*(1), 45–66.
Wang, B., Xiao, G., Yu, H., Yang, X. (2009). Distance-based outlier detection on uncertain data. In *CIT* (pp. 293–298).