A K-nearest neighbours method based on imprecise probabilities

Sébastien Destercke

A K-nearest neighbours method based on imprecise probabilities.[☆]

Sebastien Destercke[a,*]

[a]*INRA/CIRAD, UMR1208, 2 place P. Viala, F-34060 Montpellier cedex 1, France*

Abstract

K-nearest neighbours algorithms are among the most popular existing classification methods, due to their simplicity and their good performances. Over the years, several extensions of the initial method have been proposed. In this paper, we propose a K-nearest neighbours approach that uses the theory of imprecise probabilities, and more specifically lower previsions. We show that the proposed approach has several assets: it can handle uncertain data in a very generic way, and decision rules developed within this theory allow us to deal with conflicting information between neighbours or with the absence of close neighbour to the instance to classify. We show that results of the basic k-NN and weighted k-NN methods can be retrieved by the proposed approach. We end with some experiments on classical data sets.

Keywords: Classification, lower prevision, nearest neighbours, uncertain data.

1. Introduction

K-nearest neighbours (k-NN) is a well known classification method [18]. In the basic k-NN method the class of a new instance is selected as the one that is in majority among its k nearest neighbours, chosen among a set of available samples whose true class is known. This method works on the assumption that similar inputs should give similar outputs, and usually gives good classification results. Over the years, it has been applied to a number of different applications, including clustering [20], image analysis [31], data compression [21], intrusion detection [34], ...

Although the basic k-NN method results can be proved to be bounded above by twice the optimal Bayes error rate when the number of samples tends to infinity, it is not always optimal in the finite case [10]. This explains why its improvement has remained an active research topic over the years. In the recent years, most of this research has focused on how to find nearest neighbours

[☆]This paper is a revised and extended version of [13]

[*]Corresponding author

Email address: sebastien.destercke@cirad.fr (Sebastien Destercke)

efficiently, either exactly [7] or approximately [32], and on how to optimise the classification rate of the nearest neighbours procedure [39, 1, 43].

One of the main drawback of the original method is that it implicitly assumes that the k-nearest neighbours are relatively close to the instance to classify, i.e. that they are contained in a region of sufficiently small volume to provide a good estimation of conditional densities. In other words, it assumes that nearest neighbours can act as reliable information sources to estimate the unknown class. In practice, this is rarely true, and the distance between a new instance and its nearest neighbour can be large. This makes the way basic k-NN methods handle known samples questionable (giving equal weight to each neighbour and not considering whether an unknown instance is far from any known sample).

An appealing idea to solve this issue and improve the k-NN method is to weight the influence of a neighbour according to its distance to the instance to classify [16]. The class of an unknown instance is then chosen as the one for which the sum of weights of the k nearest neighbours is the greatest. However, it is not always the case that weighted k-NN can achieve better performances than basic k-NN method. Still, for a particular data set, weights can be optimised so as to achieve better performances than the basic k-NN [44]. Other authors have proposed to use different weighting schemes for samples belonging to different classes, since different classes can have different structures in the input feature space.

Yet, modifying distances and introducing weights is not sufficient to deal with other problems such as the fact that a new instance may be poorly represented in the input space (i.e., is far from any known sample) or that nearest neighbours may disagree about the output class (i.e., information about what class the unknown instance belong to is conflicting). To solve these issues, authors have respectively introduced *distance rejection* [15] or *ambiguity rejection* [8] criteria, stating that a neighbour should be considered only if it is not too far from the instance to classify, and that no class should be chosen if there is too much ambiguity about its value (i.e., conflict among neighbours).

Another limitation of basic k-NN method is that it is unable to cope with uncertainty and imprecision in the labelling of known samples. This can be a problem in practical applications where data labels are imperfectly known (for example, medical experts in diagnostic problems may find hard to affect a unique class to a sample, or sensory experts in food industry may be unable to select a precise label for some product).

To solve this issue, authors have proposed various k-NN methods based on different uncertainty models. For instance, Keller [24] has proposed a method based on fuzzy sets, while Denoeux [11] proposed a method using the more general model of belief functions. These methods allow one to model uncertainty in the sample labels and provide results comparable and sometimes better than the basic or the weighted k-NN methods. For a more detailed survey of the k-NN algorithm and its different extensions, see [22, Chap. 2].

In this paper, we propose and discuss a k-NN method based on the use of Walley's lower prevision theory [38, 25]. We will show that basing a k-NN method on lower previsions has different assets:

- as uncertainty models, lower previsions are very general. They can handle uncertainty representations that classical probabilities, fuzzy sets or belief functions cannot handle, such as comparative probabilities [28];
- cost functions naturally fits in the method, as lower previsions correspond to expectation bounds;
- the decision rules based on lower prevision can result in imprecise classification if information is judged not sufficient to provide a unique output class. This feature offers a simple way to handle both distant and conflicting nearest neighbours.

The basic material concerning lower previsions needed in this paper is given in Section 2. After recalling basic and weighted k-NN methods, Section 3 presents the method based on lower previsions. Although the method is settled in a more general uncertainty framework than the method proposed by Denoeux [11] and uses calculi proper to lower previsions, it takes many ideas from Denoeux's method, and is based on similar intuitions. In Section 4, we show that our approach encompasses basic and weighted k-NN methods as special cases, and that it naturally tackles issues mentioned above. Finally, some experiments on well-known data sets are provided in Section 5.

2. Basics of lower previsions

In the past decades, many authors have argued that probability theory alone is not able to represent faithfully all kinds of uncertainty. In particular, the validity of classical probabilistic modeling can be questioned when only a poor quantity of data is available, when there is some imprecision in the information or when one searches to model source reliability. To solve this issue, several complementary models and theories have been proposed, including robust Bayesian approach [4], possibility theory [14, 41], evidence theory [29, 30], lower previsions [38], ...

Lower previsions have been introduced by Williams [40] and formalized by Walley [38] as a unifying reasoning framework encompassing most known uncertainty models. It generalizes de Finetti's [17] subjective account of probabilities (and is therefore coherent with this latter). This section recalls the basics of lower previsions needed in this paper. We refer to [25] and [38] for more details.

2.1. Coherent lower previsions

Let Y be a variable assuming its values on a (finite) space $\mathcal{Y}$ counting N exclusive and disjoint elements. Let $\mathcal{L}(\mathcal{Y})$ denote the set of real-valued functions on $\mathcal{Y}$. A *lower prevision* $\underline{P}$ on $\mathcal{L}(\mathcal{Y})$ is defined as a mapping $\underline{P} : \mathcal{L}(\mathcal{Y}) \to \mathbb{R}$. $\underline{P}(f)$ corresponds to a lower expectation bound of the uncertain reward f and is interpreted, in Walley's subjective framework, as the supremum buying price an agent would be ready to pay for the uncertain reward f ($f(y)$ is then the reward if y turns out to be Y true value).

To a lower prevision $\underline{P}$ is associated its dual upper prevision $\overline{P}$ (or upper expectation), defined as $\overline{P}(f) = -\underline{P}(-f)$. The lower probability of an event $A \subseteq \mathcal{Y}$ corresponds to the lower prevision $\underline{P}(\mathbf{1}_{(A)})$ of the indicator function $\mathbf{1}_{(A)}$ of that event. We will simply denote it by $\underline{P}(A)$. To a lower probability is associated the dual notion of upper probability $\overline{P}(A)$ such that $\overline{P}(A) = 1-\underline{P}(\overline{A})$, where $\overline{A}$ denotes the complement of A.

A *coherent lower prevision* on $\mathcal{L}(\mathcal{Y})$ is defined as a lower prevision satisfying the following conditions:

1. $\underline{P}(f) \geq \inf_{y\in\mathcal{Y}} f(y)$ for all $f \in \mathcal{L}(\mathcal{Y})$ (accepting sure gain);
2. $\underline{P}(\lambda f) = \lambda \underline{P}(f)$ for each $f \in \mathcal{L}(\mathcal{Y})$ and $\lambda \geq 0$ (positive homogeneity);
3. $\underline{P}(f+g) \geq \underline{P}(f) + \underline{P}(g)$ for all $f, g \in \mathcal{L}(\mathcal{Y})$ (superadditivity),

and a lower prevision defined on a subset $\mathcal{K} \subseteq \mathcal{L}(\mathcal{Y})$ is called coherent if it is the restriction on $\mathcal{K}$ of some coherent lower prevision on $\mathcal{L}(\mathcal{Y})$. Note that, in this paper, we will exclusively deal with coherent lower previsions.

A coherent lower prevision on $\mathcal{L}(\mathcal{Y})$ is said to be a *linear prevision* P if it is self-conjugate, i.e., if $\underline{P}(f) = \overline{P}(f) = P(f)$ for all $f \in \mathcal{L}(\mathcal{Y})$ (in particular, $\underline{P}(A) = \overline{P}(A) = P(A)$ for any event A). A linear prevision P satisfies the property of additivity, in the sense that $P(f+g) = P(f) + P(g)$ for all $f, g \in \mathcal{L}(\mathcal{Y})$. Linear previsions correspond to the expectation operators of classical probabilities, and the set of linear previsions on $\mathcal{L}(\mathcal{Y})$ is in one-to-one correspondence with the set of probability distributions on $\mathcal{Y}$.

Consider now a coherent lower prevision $\underline{P}$ defined on some subset $\mathcal{K} \subseteq \mathcal{L}(\mathcal{Y})$. This coherent lower prevision can be associated to a convex probability set, or *credal set*, of dominating linear previsions (or probabilities). This credal set $\mathcal{P}(\underline{P})$ is defined as

$$\mathcal{P}(\underline{P}) = \{P \in \mathbb{P}_{\mathcal{Y}} | (\forall f \in \mathcal{K})(P(f) \geq \underline{P}(f))\},$$

with $\mathbb{P}_{\mathcal{Y}}$ the set of all linear previsions on $\mathcal{L}(\mathcal{Y})$. Coherent lower previsions are in one-to-one correspondence with credal sets. This provides them with a sensitivity analysis interpretation, in which $\underline{P}$ models an ill-known precise probability. This means that information about the uncertain variable Y can either be given in terms of expectation or probabilistic bounds, or in terms of sets of possible probabilities (whatever best suits the situation). Note that the value $\underline{P}(f)$ for a given f defines a linear constraint on the set of acceptable linear previsions.

When a coherent lower prevision $\underline{P}$ is defined on a subset $\mathcal{K} \subseteq \mathcal{L}(\mathcal{Y})$, its *natural extension* $\underline{E}(g)$ to some arbitrary function $g \in \mathcal{L}(\mathcal{Y})$ is defined as

$$\underline{E}(g) = \inf_{P\in\mathcal{P}(\underline{P})} P(g),$$

i.e., it is the infimum taken over expectations of g w.r.t. probability distributions dominating $\underline{P}$. This natural extension can usually be computed by using linear

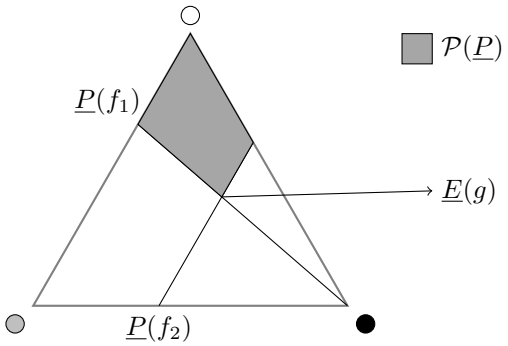


Figure 1: Lower prevision and credal set of Example 1

programming techniques, and corresponds to the most conservative inference one can make about function g if $\underline{P}$ is the only information one has. Natural extension provides an alternative definition of coherence, as a lower prevision $\underline{P}$ defined on some subset $\mathcal{K}$ is coherent if and only if $\underline{E}(g) = \underline{P}(g)$ for any $g \in \mathcal{K}$, i.e., if $\underline{P}$ is the lower envelope of $\mathcal{P}(\underline{P})$ on $\mathcal{K}$.

Example 1. *Consider a 3 element space* $\mathcal{Y} = \{●, ◍, ○\}$ *where* ● , ◍ , ○ *describe some output class of interest, for example the winner of a race having three participants. Confronted to a new race whose winner is unknown, an expert provides the following assessments:*

- $\{○\}$ *winning is at least two times more probable than* $\{◍\}$ *winning:* $2P(\{◍\}) \leq P(\{○\})$
- *the probability of* $\{●\}$ *winning is not higher than 0.4:* $P(\{●\}) \leq 0.4$

The first statement can be transformed into $0 \leq P(\{○\}) - 2P(\{◍\})$*, meaning that the lower prevision of the function* $f_1(●) = 0, f_1(◍) = -2, f_1(○) = 1$ *is such that* $\underline{P}(f_1) = 0$*. The second statement says that* $\overline{P}(\mathbf{1}_{(\{●\})}) = 0.4$*, hence that* $\underline{P}(f_2 = -\mathbf{1}_{(\{●\})}) = -0.4$ *by duality (or* $\underline{P}(\mathbf{1}_{\overline{(\{●\})}}) = 0.6$*). The coherent lower prevision* $\underline{P}$ *defined on* $\mathcal{K} = \{f_1, f_2\}$ *then models the information provided by the expert, and induces the credal set pictured*[1] *in Figure 1. According to this information, the natural extension of* $\underline{P}$ *to the function* $g(●) = -1, g(◍) = 2, g(○) = 3$ *consists in minimizing* $-1P(\{●\}) + 2P(\{◍\}) + 3P(\{○\})$ *under the constraints given by* $\underline{P}(f_1), \underline{P}(f_2)$*. The minimum is obtained for the linear prevision* $P(\{●\}) = 0.4, P(\{◍\}) = 0.2, P(\{○\}) = 0.4$*, and* $\underline{E}(g) = 0.4 \cdot -1 + 0.2 \cdot 2 + 0.4 \cdot 3 = 1.2$

[1] In the figure, each point of the simplex represents a probability distribution in barycentric coordinates, and the probability mass of an element is proportional to the distance between a point and the edge opposite to this element. See [27] for details about the simplex representation.

Coherent lower previsions are very general and encompass most known uncertainty models, including comparative probabilities [28], probability intervals [6], possibility measures [14], belief functions [29], and many other models.

In the sequel, we will repeatedly use a specific type of coherent lower prevision, called *vacuous lower prevision*. The vacuous lower prevision on an event or subset A, denoted by $\underline{P}_A$, is modeled by the single piece of information $\underline{P}_A(A) = 1$. It expresses that all is known about variable X value is that it lies in A, and nothing more. $\mathcal{P}(\underline{P}_A)$ corresponds to all probabilities that have A for support. The natural extension of $\underline{P}_A$, for any $f \in \mathcal{L}(\mathcal{Y})$, has the following simple expression:

$$\underline{E}_A(f) = \inf_{y \in A} f(y).$$

Two particular cases are of further interest: the *precise* lower prevision $\underline{P}_{\{y\}}$ on an element y is equivalent to state that the value of Y is certainly y. It is the linear prevision corresponding to the Dirac distribution on y. In this case, the natural extension to any function $g \in \mathcal{L}(\mathcal{Y})$ is $\underline{E}_{\{y\}}(g) = g(y)$; the *totally* vacuous lower prevision $\underline{P}_{\mathcal{Y}}$ corresponds to a statement of total ignorance about the value of Y (outside the fact that it lies in $\mathcal{Y}$).

2.2. Vacuous mixture and lower previsions merging

Further on we will consider the different neighbours as information sources that provide non-totally reliable information (modelled by coherent lower previsions) about the class of an unknown instance. We therefore need proper tools to handle these different pieces of information and to obtain a unique coherent lower prevision from them. To do so, we need:

- a means to take account of the source reliability, weakening the provided information accordingly to this reliability,
- a rule to merge the different lower previsions and their natural extensions into a single one, representing our final beliefs.

Given a coherent lower prevision $\underline{P}$, its natural extension $\underline{E}$ and a scalar $\epsilon \in [0, 1]$, the (coherent) lower prevision $\underline{P}^\epsilon$ that we call here *ϵ-vacuous mixture* is such that $\underline{P}^\epsilon = \epsilon\underline{P} + (1 - \epsilon)\underline{P}_{\mathcal{Y}}$. Its natural extension $\underline{E}^\epsilon$ for any $f \in \mathcal{L}(\mathcal{Y})$ is such that

$$\underline{E}^\epsilon(f) = \epsilon\underline{E}(f) + (1 - \epsilon)\underline{E}_{\mathcal{Y}}(f).$$

ϵ can be interpreted as the reliability of the information $\underline{P}$, $1 - \epsilon$ being then the probability of knowing nothing (i.e., being ignorant). ϵ-vacuous mixtures constitute a simple means to take account of source reliability. This reliability ϵ can be, for example, a decreasing function of the distance between an instance and a nearest neighbour (see Section 3).

Example 2. *Consider again Example 1 and a discounting factor $\epsilon = 0.7$. The vacuous mixture with the lower prevision of Example 1 is such that its natural*

extension to function g is

$$\underline{E}^{\epsilon} = \epsilon \underline{E}(g) + (1-\epsilon) \inf_{y \in \mathcal{Y}} g(y) = 0.7 \cdot 1.2 + 0.3 \cdot -1 = 0.54$$

Now, consider k coherent lower previsions $\underline{P}_1, \ldots, \underline{P}_k$ and their natural extensions $\underline{E}_1, \ldots, \underline{E}_k$, provided for example by the k nearest neighbours of an instance to classify. To build a single coherent lower prevision from these ones, we propose to use a merging rule that consists in taking the arithmetic mean of the different lower previsions (or, equivalently, of their natural extensions). That is, we compute $\underline{E}_\Sigma$ such that $\underline{E}_\Sigma(f) = \frac{1}{k} \sum_{i=1}^{k} \underline{E}_i(f)$ for any $f \in \mathcal{L}(\mathcal{X})$. This rule has been justified and used by different authors to merge coherent lower previsions (or credal sets) [37, 26]. It has two advantages: the arithmetic mean of coherent lower previsions is always a coherent lower prevision (while other merging rules such as conjunctive ones may result in lower previsions inducing empty credal sets, i.e. lower previsions not accepting sure gain), and it is easy to compute.

This rule also appears to be the most coherent with classical (weighted) k-NN methods, since these methods can be interpreted as non-parametric density estimation techniques that consists in taking the (weighted) arithmetic mean of Dirac distributions. Estimated density is then used to choose the instance class. Actually, we will show in Section 4 that using this merging rule with specific parameters allows us to retrieve classical (weighted) k-NN methods as special cases.

2.3. Decision rules

When considering an imperfectly known variable Y, choosing its class comes down to decide its value, given the information we have about the variable. In some cases, one can also have some information about the costs of choosing a wrong class instead of the true one. Here we assume that these costs are modelled for each $y \in \mathcal{Y}$ by positive bounded real-valued functions $c_y : \mathcal{Y} \to \mathbb{R}^+$, where $c_y(x)$ is the cost of selecting x when y is the true class. Reward functions can then be retrieved by considering functions $-c_y$.

When uncertainty over $\mathcal{Y}$ is represented by a linear prevision P, the chosen class $\widehat{y}$ is usually the one that has the highest expected gain (or lowest expected cost), i.e. $\widehat{y} = \arg\max_{y \in \mathcal{Y}} E(-c_y)$, with E the natural extension of P. When knowledge about Y value is given as a lower prevision $\underline{P}$, expected gain (or cost) becomes imprecise, as lower and upper expectations do no longer coincide. Hence, the classical decision rule used for linear prevision has to be extended to handle this situation [33]. There are two main ways to do so.

The first way is to still require the decision to be a single element, that is to provide only one class as final answer. The most well-known decision rule in this category is the maximin rule [19], for which the final decision is such that

$$\widehat{y} = \arg\max_{y \in \mathcal{Y}} \underline{E}(-c_y).$$

This amounts to maximising the lower expected gain, i.e., to be pessimistic and take the best outcome in the worst possible situation. Other possible rules of this kind include maximising the upper expected gain (corresponding to an optimistic or "bold" attitude) or maximising a value in-between [23]. In this paper, the maximin rule is retained when a precise classification is needed, for it corresponds to a cautious attitude.

The other way to extend the classical decision rule is to provide a set (possibly, but not necessarily reduced to a singleton) of optimal classes. The idea behind such extensions is that our information may be insufficient to select only one class, the number of selected optimal classes then reflecting the imprecision in our information. This requires to build among the possible decisions (here, the classes) a partial ordering instead of a complete one, and then to select only the optimal decisions, that is those that are not dominated by another one. Here we consider interval ordering $\leq_I$ for two reasons: it is computationally convenient and it corresponds again to a cautious attitude, as the set of optimal classes selected by this rule includes the sets built by other rules [33, 12].

The interval ordering is such that a decision y is dominated by a decision x, denoted by $y \leq_I x$, iff $\overline{E}(-c_y) \leq \underline{E}(-c_x)$, that is if the upper expected gain of picking y is sure to be lower than the lower expected gain of picking x. The decision set $\widehat{Y}$ is then

$$\widehat{Y} = \{y \in \mathcal{Y} |\ \nexists x \in \mathcal{Y} \text{ s.t. } y \leq_I x\}.$$

When $\widehat{Y}$ contains more than one element, we will speak of imprecise classification.

Example 3. *Consider again Example 1 and the following cost functions*

$$c_{\bigcirc}(\bigcirc) = 0, c_{\bigcirc}(\textcolor{gray}{\bullet}) = 2, c_{\bigcirc}(\bullet) = 2; \qquad c_{\bullet}(\bigcirc) = 3, c_{\bullet}(\textcolor{gray}{\bullet}) = 1, c_{\bullet}(\bullet) = 0;$$

$$c_{\textcolor{gray}{\bullet}}(\bigcirc) = 1, c_{\textcolor{gray}{\bullet}}(\textcolor{gray}{\bullet}) = 0, c_{\textcolor{gray}{\bullet}}(\bullet) = 2.$$

The lower and upper natural extensions of the corresponding reward or utility functions, given the lower coherent prevision $\underline{P}$ of Example 1, are

$$[\underline{E}(-c_{\bigcirc}), \overline{E}(-c_{\bigcirc})] = [-1.2, 0]; [\underline{E}(-c_{\bullet}), \overline{E}(-c_{\bullet})] = [-3, -1.4];$$

$$[\underline{E}(-c_{\textcolor{gray}{\bullet}}), \overline{E}(-c_{\textcolor{gray}{\bullet}})] = [-1.4, -2/3].$$

The maximin and interval dominance rules respectively provide the optimal sets $\widehat{y} = \bigcirc$ and $\widehat{Y} = \{\bigcirc, \textcolor{gray}{\bullet}\}$.

3. Introducing the *k*-NN methods

In this section, we briefly recall the basic and weighted k-NN methods, before introducing the method based on lower previsions.

We use the following notation: $\vec{x}_1, \ldots, \vec{x}_N$ are the input values of N known sample, with $\vec{x}_i \in \mathbb{R}^D$ a D-dimensional vector, and $d : \mathbb{R}^D \times \mathbb{R}^D \to \mathbb{R}$ is a

distance between such vectors (here, the Euclidean distance). $\mathcal{Y} = \{y_1, \ldots, y_M\}$ is the set of possible classes, and $\underline{P}_i : \mathcal{L}(\mathcal{Y}) \to \mathbb{R}$ is the lower prevision modelling the knowledge about the class to which the sample $\vec{x}_i$ belongs.

3.1. Basic and weighted k-NN

In the basic methods, classes of known samples are assumed to be perfectly known. In this case, the lower prevision $\underline{P}_i$ is the precise lower prevision $\underline{P}_{\{y_i\}}$, with y_i the class of sample $\vec{x}_i$.

Given a new instance $\vec{x}$ to classify, denote by $\vec{x}_{(1)}, \ldots, \vec{x}_{(k)}$ its k ordered nearest neighbours (i.e. $d_{(i)} < d_{(j)}$ if $i \leq j$, with $d_{(i)} := d(\vec{x}, \vec{x}_{(i)})$). The basic k-NN method consists in choosing the class $\widehat{y}$ of $\vec{x}$ as the class in majority among its k nearest neighbours, i.e.,

$$\widehat{y} = \arg\max_{y \in \mathcal{Y}} \sum_{i=1,\ldots,k} \underline{E}_{\{y_{(i)}\}}(\mathbf{1}_{(y)}), \tag{1}$$

with $\underline{E}_{\{y_{(i)}\}}(\mathbf{1}_{(y)}) = 1$ if $y = y_{(i)}$, zero otherwise.

In the weighted version of the k-NN method, weights $\epsilon_{(i)}$ that are proportional to $d_{(i)}$ are associated to each nearest neighbours. The class $\widehat{y}$ of $\vec{x}$ is then the one whose cumulated weight is the highest, i.e.,

$$\widehat{y} = \arg\max_{y \in \mathcal{Y}} \sum_{i=1,\ldots,k} \epsilon_{(i)} \underline{E}_{\{y_{(i)}\}}(\mathbf{1}_{(y)}). \tag{2}$$

Note that both $\sum_{i=1,\ldots,k} \underline{E}_{\{y_{(i)}\}}$ and $\sum_{i=1,\ldots,k} \epsilon_{(i)} \underline{E}_{\{y_{(i)}\}}$ are linear previsions (i.e. precise probabilities) that correspond to estimated conditional probabilities obtained by usual k-NN methods. For more classical descriptions and exposure of usual k-NN methods, we refer to [22, Chap. 2].

3.2. Method based on lower previsions

Let us now consider the case where $\underline{P}_{(i)}$ is a generic coherent lower prevision on $\mathcal{L}(\mathcal{Y})$. For a given nearest neighbour $\vec{x}_{(i)}$, the knowledge $\underline{P}_{(i)}$ can be regarded as a piece of evidence related to the unknown class of $\vec{x}$. However, this piece of knowledge is not 100% reliable, and should be discounted by a value $\epsilon_{(i)} \in [0, 1]$ using a ϵ-vacuous mixture. $\underline{P}_{(i)}$ and its natural extension $\underline{E}_{(i)}$ then become, for any $f \in \mathcal{L}(\mathcal{Y})$,

$$\underline{E}_{(i)}^{\epsilon_{(i)}}(f) = \epsilon_{(i)} \underline{E}_{(i)} + (1 - \epsilon_{(i)}) \inf_{y \in \mathcal{Y}} f(y).$$

It seems natural to ask for $\epsilon_{(i)}$ to be a decreasing function of $d_{(i)}$, since the further away is the neighbour, the less reliable is the information it provides about the unknown class. Similarly to Denoeux [11], we propose to adopt the general formula

$$\epsilon_{(i)} = \epsilon_0 \phi(d_{(i)}),$$

where ϕ is a non-increasing function that can depend on the class given by $\vec{x}_{(i)}$ (if this latter one is precisely known). In addition, the following conditions

should hold:

$$0 < \epsilon_0 < 1,$$

$$\phi(0) = 1 \text{ and } \lim_{d \to \infty} \phi(d) = 0.$$

The first condition implies that even if the new instance has the same input as one training data sample, we do not consider it to be 100% reliable, as the relation linking the input feature space and the output classes is not necessarily a function.

The different (discounted) natural extensions $\underline{E}_{(1)}^{\epsilon_{(1)}}, \ldots, \underline{E}_{(k)}^{\epsilon_{(k)}}$ are then combined into a global coherent lower prevision

$$\underline{E}_{\vec{x}} = \frac{1}{k} \sum_{i=1}^{k} \underline{E}_{(i)}^{\epsilon_{(i)}}.$$

Using $\underline{E}_{\vec{x}}$ as the final uncertainty model for the true class of $\vec{x}$ and the cost function associated to each class, one can predict $\vec{x}$ final class, either as a single class by using a maximin-like criteria or as a set of possible classes by using interval dominance.

Example 4. *Consider the same space $\mathcal{Y}$ as in Example 1, and the following information about a new instance $\vec{x}$ coming from its three nearest neighbours:*

- *the first neighbour has been observed with certainty as ○, thus $\underline{P}_{(1)} = \underline{P}_{\{○\}}$;*
- *the second neighbour has been observed as ● by a device that has 80% reliability, thus $\underline{P}_{(2)}$ can be summarized by the constraint $\underline{P}(\{●\}) = 0.8$;*
- *finally, the third neighbour information has been given by an expert, who believes that the third neighbour class is ○, but is quite unsure about it. He/she has given the following information $\underline{P}_{(3)}$: $\underline{P}(\{○\}) = 0.4, \underline{P}(\{●\}) = 0.25, \underline{P}(\{◍\}) = 0.25$.*

Distances of the three neighbours are $\epsilon_{(1)} = \phi(d_{(1)}) = 0.9$, $\epsilon_{(2)} = 0.8$, $\epsilon_{(3)} = 0.6$. The obtained final lower prevision $\underline{P}_{\vec{x}}$ is therefore $\underline{P}_{\vec{x}} = 1/3\underline{P}_{(1)}^{0.9} + 1/3\underline{P}_{(2)}^{0.8} + 1/3\underline{P}_{(3)}^{0.6}$. We consider that the cost function for an element $y \in \mathcal{Y}$ is the unitary cost function (that is, the function that has value 0 on y and 1 otherwise). Maximin and interval dominance decision rules on such functions comes down to compare lower and upper probabilities of each element. Results are summarised in Table 1. Given these values, the maximin and interval dominance rules would give $\widehat{y} = ○$ and $\widehat{Y} = \{○, ●\}$, respectively. So, while ○ would be chosen as the answer if precision is required, ● cannot be totally discarded from the possible classes.

3.3. Using lower previsions to choose k

A problem when using the k-NN methods is to choose the "best" number k of neighbours to consider. This number is often selected as the one achieving

	{○}	{◍}	{●}
$\overline{E}_{\vec{x}}$	0.687	0.357	0.57
$\underline{E}_{\vec{x}}$	0.380	0.050	0.263

Table 1: Lower/upper probabilities obtained by natural extension of $\underline{P}_{\vec{x}}$ of Example 4

	$k=1$	$k=2$	$k=3$
$[\underline{E}_{\vec{x}}(\{\bullet\}), \overline{E}_{\vec{x}}(\{\bullet\})]$	$[0.9, 1]$	$[0.45, 0.6]$	$[0.3, 0.533]$
$[\underline{E}_{\vec{x}}(\{\circ\}), \overline{E}_{\vec{x}}(\{\circ\})]$	$[0, 0.1]$	$[0.4, 0.55]$	$[0.466, 0.7]$
$[\underline{E}_{\vec{x}}(\{◍\}), \overline{E}_{\vec{x}}(\{◍\})]$	$[0, 0.1]$	$[0, 0.15]$	$[0, 0.233]$
$\widehat{Y}$	$\{\bullet\}$	$\{\bullet, \circ\}$	$\{\bullet, \circ\}$
$WKNN$	●	●	○

Table 2: Evolution lower/upper probabilities of Example 5 and decisions obtained by using interval dominance rule ($\widehat{Y}$) and weighted nearest neighbour (WKNN).

the best performance in a cross-validation procedure, but k-NN methods are known to display erratic performances if k is slightly increased or decreased, even if it is by one. The next example shows that using lower previsions and imprecise decision rule can make changes in decisions less severe.

Example 5. *Again, consider the same space $\mathcal{Y}$ as in Example 1. Assume that, for an instance $\vec{x}$, the three nearest neighbours provide precise information such that $\underline{P}_{(1)} = \underline{P}_{\{\bullet\}}$, $\underline{P}_{(2)} = \underline{P}_{\{\circ\}}$ and $\underline{P}_{(3)} = \underline{P}_{\{\circ\}}$. Distances of the three neighbours are $\epsilon_{(1)} = \phi(d_{(1)}) = 0.9$, $\epsilon_{(2)} = 0.8$, $\epsilon_{(3)} = 0.6$. Table 2 summarises the evolution of lower and upper probabilities and the decisions obtained by both interval dominance and the classical weighted nearest neighbour method.*

We can see that by using lower previsions and an imprecise decision rule, the change in the decision is less abrupt: decision shift from $\{\bullet\}$ (k=1) to $\{\bullet, \circ\}$ (k=3), instead of suddenly changing from $\bullet$ to $\circ$ in the case of the weighted nearest neighbour method.

Note that, even with imprecise decision rule, significant changes in the decision can happen when k is decreased or increased. Indeed, if $\epsilon_{(3)} = 0.75$ in Example 5, then $\widehat{Y} = \{\bullet\}$ for $k = 3$. That is, decision can change if strong evidence in favour of such a change is provided by close neighbours. However, by allowing imprecise decision such changes will be less sensible to small modifications of k.

Another feature of the method is that the imprecision of $\underline{E}_{\vec{x}}$ will increase with the number k (as shown by the increasing width of the intervals in Table 2). This means that, roughly speaking, increasing k will generally (but not always) increase the number of optimal classes selected by interval dominance. So, in general, higher k values will produce more robust but less informative results.

This suggests that, when using lower previsions and imprecise decision rules, k value should be settled so that results present the "best" balance between classification accuracy and classification precision. To do so, we propose a new approach for choosing k, using the notion of *discounted accuracy* coming from multi-label classification, whose relevance to assess the quality of imprecise probabilistic classifiers has been recently discussed by Zaffalon *et al.* [42]. Consider T test samples and denote by $\widehat{Y}_i$ the decision set obtained by the method and the selected decision rule for the i^{th} test sample. The *discounted accuracy* is then [35, 9]

$$d-acc = \frac{1}{T}\sum_{i=1}^{T}\frac{\Delta_i}{f_{acc}(|\widehat{Y}_i|)}, \tag{3}$$

with $\Delta_i : 2^{|\mathcal{Y}|} \to \{0,1\}$ the function such that $\Delta_i = 1$ if y_i, the class of the i^{th} test sample, is in $\widehat{Y}_i$ and 0 otherwise. That is Δ_i has value one if the right answer is in the set of optimal classes. f_{acc} is an increasing function discounting the accuracy value by the imprecision of the classification. f_{acc} can be defined accordingly to the relative importance one wants to give to precision and accuracy. For instance, choosing the identity map $f_{acc}(|\widehat{Y}_i|) = |\widehat{Y}_i|$ comes down to consider that two classifiers CL_1 and CL_2 are equivalent (according to $d-acc$) in the following situation: CL_1 gives only precise classes, and is accurate on $T/2$ instances; CL_2 is accurate for the T instances, but provide 2 classes for each. Choosing a convex function (e.g.,$f_{acc}(|\widehat{Y}_i|) = |\widehat{Y}_i|^2$) will give more importance to precision, while a concave one (e.g., $f_{acc}(|\widehat{Y}_i|) = \sqrt{|\widehat{Y}_i|}$) will give more importance to accuracy. Note also that the cardinality of $\mathcal{Y}$, i.e., the number of possible classes should be taken into account when choosing f_{acc}. Indeed, a classifier providing 2 classes as possible output for a given instance is far more informative when it is 2 classes out of 26 than when it is 2 classes out of 2.

If $d-acc_{(k)}$ denotes the discounted accuracy obtained by considering k nearest neighbours, then we propose to define the optimal number k^* of neighbours as

$$k^* = \arg\max_{k\in\mathbb{N}} d-acc_{(k)}.$$

That is, the number having the best discounted-accuracy. In practice, the number of nearest neighbours to consider should remain low (k should be limited to values far lower than N). The idea of this rule is to choose the value k^* achieving the best compromise between informativeness and accuracy (similarly to some expert evaluation methods used in classical probabilities [3]).

4. Method properties

In this section, we first show that basic and weighted k-NN methods are embedded in our approach. We then illustrate that considering a lower prevision approach with an imprecise decision rule naturally tackles the problem of distant neighbours and of conflicting information. Note that uncertain labels are

handled in a straightforward way, thanks to the generality of lower previsions (see Section 2).

4.1. Precise training samples and unitary costs

We now investigate the specific (but classical) case where each known sample $\vec{x}_i$ has a precisely known class y_i and the costs functions are unitary, that is for any $y \in \mathcal{Y}$, $c_y(y) = 0$ and 1 otherwise. We show that in this specific case, our approach can retrieve classical results from basic and weighted k-NN methods.

First note that, given these cost functions and any coherent lower prevision $\underline{P}$ on $\mathcal{L}(\mathcal{Y})$, the lower expectation for $-c_y$ is

$$\underline{E}(-c_y) = \underline{E}(1 - c_y) - 1 = \underline{E}(\{y\}) - 1 = \overline{E}(\overline{\{y\}}),$$

that is the upper probability of $\overline{y}$. Let us first deal with the basic k-NN method.

Proposition 1. *Let k be the number of nearest neighbours considered. If training samples are precise, costs unitary and discounting rates $\epsilon_{(1)} = \ldots = \epsilon_{(k)} = \epsilon$, then the method used with a maximin decision criteria gives the same result as the basic k-NN rule, given by Eq* (1)*.*

Proof Consider a given $y \in \mathcal{Y}$ and its unitary cost function c_y. Let us now compute the natural extension $\underline{E}_{\vec{x}}$ to $-c_y$. Given the k nearest neighbours, whose lower previsions $\underline{P}_{\{y_{(i)}\}}$ are precise lower previsions, this natural extension is given by

$$\begin{aligned}\underline{E}_{\vec{x}}(-c_y) &= \frac{1}{k}\sum_{i=1}^{k} \epsilon_{(i)} \underline{E}_{\{y_{(i)}\}}(-c_y) + (1 - \epsilon_{(i)}) \inf_{y' \in \mathcal{Y}} c_y(y') \\ &= \frac{\epsilon}{k}\sum_{i=1}^{k} \underline{E}_{\{y_{(i)}\}}(\{y\}) - 1 = -\epsilon + \frac{\epsilon}{k}\sum_{i=1}^{k} \underline{E}_{\{y_{(i)}\}}(\{y\}),\end{aligned}$$

since $\inf_{y' \in \mathcal{Y}} c_y(y') = 0$ and all $\epsilon_{(i)}$ are equal. Taking the maximin rule, we obtain

$$\begin{aligned}\widehat{y} &= \arg\max_{y \in \mathcal{Y}} \underline{E}_{\vec{x}}(-c_y) \\ &= \arg\max_{y \in \mathcal{Y}} \left(\epsilon + \frac{\epsilon}{k}\sum_{i=1}^{k} \underline{E}_{\{y_{(i)}\}}(\{y\}) \right) = \epsilon + \frac{\epsilon}{k} \arg\max_{y \in \mathcal{Y}} \left(\sum_{i=1}^{k} \underline{E}_{\{y_{(i)}\}}(\{y\}) \right) \\ &= \arg\max_{y \in \mathcal{Y}} \left(\sum_{i=1}^{k} \underline{E}_{\{y_{(i)}\}}(\{y\}) \right).\end{aligned}$$

The last line coinciding with Eq (1), the proof is finished. □

Note that this result asserts that, asymptotically, the performances of the proposed method are bounded by twice the optimal Bayes rate error if one chooses equal weights together with a maximin rule. Let us now show that the

results of the weighted k-NN can also be retrieved as a particular case of our approach.

Proposition 2. *Let k be the number of nearest neighbours considered. If training samples are precise, costs unitary and discounting rates $\epsilon_{(i)}$, then the method used with a maximin decision criteria gives the same result as a weighted k-NN rule with the weights $\epsilon_{(i)}$, given by Eq* (2)*.*

Proof The natural extension $\underline{E}_{\vec{x}}(-c_y)$ to $-c_y$ for some $y \in \mathcal{Y}$ becomes

$$\begin{aligned}\underline{E}_{\vec{x}}(-c_y) &= \frac{1}{k}\sum_{i=1}^{k} \epsilon_{(i)} \underline{E}_{\{y_{(i)}\}}(-c_y) + (1-\epsilon_{(i)}) \inf_{y' \in \mathcal{Y}} c_y(y') \\ &= \frac{1}{k}\sum_{i=1}^{k} \epsilon_{(i)} (\underline{E}_{\{y_{(i)}\}}(\{y\}) - 1) = \frac{-\sum_{i=1}^{k} \epsilon_{(i)}}{k} + \frac{1}{k}\sum_{i=1}^{k} \epsilon_{(i)} (\underline{E}_{\{y_{(i)}\}}(\{y\}).\end{aligned}$$

Using a maximin decision rule, we get

$$\begin{aligned}\widehat{y} &= \arg\max_{y \in \mathcal{Y}} \left(\frac{-\sum_{i=1}^{k} \epsilon_{(i)}}{k} + \frac{1}{k}\sum_{i=1}^{k} \epsilon_{(i)} (\underline{E}_{\{y_{(i)}\}}(\{y\}) \right) \\ &= \frac{-\sum_{i=1}^{k} \epsilon_{(i)}}{k} + \frac{1}{k} \arg\max_{y \in \mathcal{Y}} \left(\sum_{i=1}^{k} \epsilon_{(i)} (\underline{E}_{\{y_{(i)}\}}(\{y\}) \right) \\ &= \arg\max_{y \in \mathcal{Y}} \left(\sum_{i=1}^{k} \epsilon_{(i)} (\underline{E}_{\{y_{(i)}\}}(\{y\}) \right).\end{aligned}$$

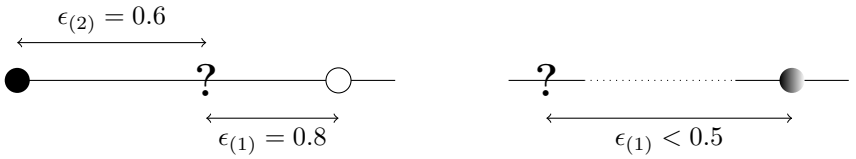
Since this result coincides with Eq. (2), the proof is finished. □

Note that these two propositions mean that the proposed approach can use the improvements proposed for both the basic and weighted k-NN methods to improve its own results (such as the use of kernels).

4.2. Distant neighbours and conflicting information

Using interval dominance is a good way to treat both ambiguity between the nearest neighbours and large distances between the instance to classify and its nearest neighbour. Indeed, if all nearest neighbours agree on the output class and are close to the new instance, the obtained lower prevision $\underline{P}_{\vec{x}}$ and its natural extension will be precise enough so that $\widehat{Y}$ includes only one optimal class. On the contrary, if nearest neighbours disagree or are far from the new instance, $\underline{P}_{\vec{x}}$ will be imprecise and $\widehat{Y}$ will include several classes. Let us illustrate this on two short examples.

Example 6. *Assume we have the situation pictured in Figure 2.a, where $k = 2$ and the two nearest neighbours are exactly known. For simplicity, assume costs are unitary. Using interval dominance or maximin in such a case comes down*



(a) conflicting neighbours (b) distant neighbours

Figure 2: Conflicting and distant neighbours issues.

to compare natural extension over indicator functions of singletons, and in this case

$$[\underline{E}(\{\bigcirc\}), \overline{E}(\{\bigcirc\})] = [0.4, 0.7]; \quad [\underline{E}(\{\bullet\}), \overline{E}(\{\bullet\})] = [0.3, 0.6];$$
$$[\underline{E}(\{\circledcirc\}), \overline{E}(\{\circledcirc\})] = [0, 0.25],$$

and the set of optimal classes given by interval dominance is $\widehat{Y} = \{\bullet, \bigcirc\}$.

Example 7. *Consider now the situation in Figure 2.b. The color of the first neighbour has no importance, but for the sake of the example let us assume that it is white and that* $\epsilon_{(1)} = 0.5 - \delta$*, with* δ *any small positive value. In such a case, and assuming unitary costs, we have*

$$[\underline{E}(\{\bigcirc\}), \overline{E}(\{\bigcirc\})] = [0.5-\delta, 1]; \quad [\underline{E}(\{\bullet\}), \overline{E}(\{\bullet\})] = [\underline{E}(\{\circledcirc\}), \overline{E}(\{\circledcirc\})] = [0, 0.5+\delta].$$

The set of optimal classes given by interval dominance is here $\widehat{Y} = \mathcal{Y}$*, i.e., total undecidability due to too distant neighbours.*

Examples 6 and 7 also illustrate that it may be difficult to recognize whether imprecise classification comes from conflict or distance. In practice, in the latter case the decision rules will consider all the possible classes as optimal (due to the importance given to the vacuous lower prevision), while in the former case those same rules will often consider only some classes as optimal, excluding the others. Although this kind of rough differentiation can work when the number of possible classes is high, it will usually be useless when the number of classes is low (consider in particular the case of two classes, where any imprecision in the decision means considering all the classes as optimal, whatever the source of the imprecision is).

Building on Example 7, we propose an easy rule to differentiate the two situations: imprecise classification of an instance $\vec{x}$ will be declared due to distance if $\epsilon_{(1)} < 0.5$, for the reason that with precise labels and unitary costs, $\epsilon_{(1)} < 0.5$ is enough to ensure that $\widehat{Y} = \mathcal{Y}$.

5. Experiments

Propositions 1 and 2 show that usual k-NN methods can be retrieved by our approach, and that proposed improvements of distance and weight computation

Name	# instances	# input variables	# output classes
Balance scale	625	4	3
Glass	214	9	6
Image segmentation	2100	19	7
Ionosphere	351	9	2
Iris	150	4	3
Letter recognition	3000	16	26
Liver disorder	345	7	2
Pima Diabetes	768	8	2
Wine	178	13	3
Yeast	1484	8	10
Zoo	101	17	7

Table 3: Experiment data sets

can be easily integrated in them. We can thus reach performances of k-NN methods existing in the literature (see, e.g., [39, 43]).

The experiments presented in this section mainly compare the result of our approach on classical data sets with the results of classical k-NN methods (either the basic method or the weighted k-NN with weights equivalent to computed $\epsilon_{(i)}$), using the discounted accuracy to do so.

5.1. Experiment set-up

We selected 11 data sets from the UCI repository [2]. For all of them, input variables take either real or integer values, and output variables consists in a finite number of classes. They are listed and summarized in Table 3. The validation method used in the experiments is a leave-one-out cross-validation.

To estimate the accuracy, we have considered $f_{acc}(|\widehat{Y}_i|) = \sqrt{|\widehat{Y}_i|}$ in Eq. (3), so that imprecise classifications are not too penalized. We have also computed undiscounted accuracy (that is, Eq. (3) with $f_{acc} = 1$).

The Euclidean distance is used to compute distance $d(\vec{x}, \vec{x}_i)$ between an instance $\vec{x}$ and the ith sample. As discussing and optimising ϕ is not the topic of the paper, we consider the simple heuristic[11] where, for a given test data $\vec{x}$,

$$\phi(d_{(i)}) = \exp^{-d^{\beta}_{(i)}/\overline{d}_{y_{(i)}}},$$

with $\overline{d}_{y_{(i)}}$ the average distance between elements of the training set having $y_{(i)}$ for class, and β a parameter that we fix to $\beta = 1.5$, so that ϕ increase in a moderate way (neither too slowly nor too quickly). For the same reason, we fix $\epsilon_0 = 0.99$, so that

$$\epsilon_{(i)} = 0.99 \exp^{-d^{1.5}_{(i)}/\overline{d}_{y_{(i)}}},$$

with the same weights being used in the weighted k-NN method. For each data set, we let k vary from 1 to 10.

Data set	$d-acc$				$\#\ \lvert\widehat{Y}\rvert > 1$ dist.
	KNN	WKNN	LPKNN	$f_{acc}=1$	neighbours
Balance scale	0.89 (10)	**0.90 (10)**	0.89 (10)	0.93 (10)	0
Glass	0.73 (1)	0.74 (2)	**0.74 (2)**	0.82 (2)	2
Image segmentation	0.96 (1)	0.96 (1)	**0.96 (2)**	0.98 (2)	0
Ionosphere	**0.9 (2)**	0.9 (2)	0.89 (6)	0.93 (6)	6
Iris	**0.97 (4)**	0.97 (4)	0.96 (1)	0.99 (8)	4
Letter recognition	0.88 (1)	0.88 (1)	**0.88 (2)**	0.92 (2)	0
Liver disorder	0.69 (9)	0.69 (9)	**0.71 (8)**	0.81 (2)	0
Pima Diabetes	0.73 (9)	0.73 (9)	**0.75 (10)**	0.86 (2)	0
Wine	0.77 (1)	0.77 (1)	**0.78 (2)**	0.86 (2)	0
Yeast	**0.59 (10)**	0.58 (9)	0.52 (3)	0.94 (10)	162
Zoo	**0.98 (1)**	0.98 (1)	0.95 (1)	1 (1)	8

Table 4: Experiment results for basic KNN (KNN), weighted KNN (WKNN) and this paper approach (LP-KNN). Number of neighbours associated to $d-acc$ are between parenthesis.

5.2. Results and discussion

Table 4 summarises the results obtained for each data set:

- The best value of the discounted accuracy for the basic k-NN (KNN), the weighted k-NN (WKNN) and our own approach (LPKNN), as well as the number of neighbours for which this best value has been obtained (in parenthesis);
- The best value of the undiscounted accuracy ($f_{acc}=1$), as well as the number of neighbours for which this best value has been obtained (in parenthesis);
- The number of imprecise classifications due to distant neighbours.

Best obtained classification rates in Table 4 are in bold (excluding performances obtained with $f_{acc}=1$). From these results, it appears that the imprecise classifier based on lower previsions competes quite well. In most cases, it reaches the same performances as the basic k-NN methods, and outperforms them in three cases.

However, having the same performances in terms of discounted accuracy does not mean having the same rate of well-classified items. This is shown by the difference between the column $f_{acc}=1$ (where imprecision is not discounted) and the LPKNN column. Indeed, the greater the difference between these two column results, the more imprecision there is in LPKNN. This shows, among other things, that including some imprecision in the decision and the uncertainty modeling can provide more robust results without deteriorating the performances (take for instance the Letter recognition data set, in which including one more neighbour allow to better recognize the letters, and to provide multiple letters when the class of one instance is unclear).

The fact that there is sometimes a significative difference between LPKNN and $f_{acc} = 1$ performances can have two main explanations:

- nearest neighbours provide conflicting information about the instance class. Such data sets have usually no or very few imprecise classification due to distant neighbours (it is the case, for instance, for data sets Pima or Wine);
- the data set is too sparse and many or some instances to classify have no neighbour near them. Such data sets can be recognized by the fact that the number of imprecise classifications due to distant neighbours is high (it is the case, for instance, of the Zoo and Yeast data sets).

In each case, this gap indicates that classification results could probably be improved by taking specific actions, for example by performing some feature selection (to reduce the input space dimension or to remove noisy features), by weighting the different features according to their importance or by removing some atypical examples.

6. Conclusion and perspectives

In this paper, we have defined a k-NN method based on lower previsions. As lower previsions are very generic models of uncertainty, they allow to handle a great variety of uncertainty representations such as classical probabilities, possibility distributions (fuzzy sets), belief functions or comparative probabilities. Using lower previsions also allows us to solve the issues of ambiguity (conflicting information) and of absence of neighbours close to a given instance by simply using decision rules proposed by the theory. Such decision rules select a set of possible (i.e., optimal) classes rather than a single one when information delivered by neighbours is ambiguous or unreliable.

Using this particular feature of lower previsions, we have proposed a simple and new means to select the "best" number k of nearest neighbours to consider. To do this, we use the notion of discounted accuracy, so that a good balance is achieved between classification precision and accuracy. Experimental results on several benchmark data sets have shown that introducing imprecision in the classification does not lower the classifier performances, but gives more robust results and valuable insights about the considered data sets.

This paper has exposed the basics of a k-NN method using lower previsions. Many related topics remain to be investigated, among which:

- how to optimise the whole procedure so that it can give better results for a given problem? This could be done, for instance, by optimizing parameter values [44] or by weighting the different input features.
- how does the method compare to another approach that would consist in starting from a vacuous multinomial model such as the Imprecise Dirichlet Model [5] and updating this model with (discounted) information coming from the neighbours [36].

- how the framework of lower previsions can help in solving the problem of instances having uncertain / missing input values?

References

[1] J. Amores, N. Sebe, P. Radeva, Boosting the distance estimation: Application to the k-nearest neighbor classifier, Pattern Recognition Letters 27 (2006) 201–209.

[2] A. Asuncion, D. Newman, UCI machine learning repository [http://www.ics.uci.edu/∼mlearn/MLRepository.html], 2007.

[3] T. Bedford, R. Cooke, Probabilistic Risk Analysis. Foundations and Methods, Cambridge University Press, UK, 2001.

[4] J.O. Berger, An overview of robust Bayesian analysis, Test 3 (1994) 5–124. With discussion.

[5] J.M. Bernard, An introduction to the imprecise dirichlet model, Int. J. of Approximate Reasoning 39 (2008) 123–150.

[6] L. de Campos, J. Huete, S. Moral, Probability intervals: a tool for uncertain reasoning, I. J. of Uncertainty, Fuzziness and Knowledge-Based Systems 2 (1994) 167–196.

[7] Y.S. Chen, Y.P. Hung, T.F. Yen, C.S. Fuh, Fast and versatile algorithm for nearest neighbor search based on a lower bound tree, Pattern Recognition 40 (2007) 360–375.

[8] C. Chow, On optimum recognition error and reject tradeoff, IEEE Trans. Inform. Theory 16 (1970) 21–27.

[9] G. Corani, M. Zaffalon, Lazy naive credal classifier, in: KDD Workshop on Knowledge Discovery from Uncertain Data, pp. 30–37.

[10] T. Cover, P. Hart, Nearest neighbor pattern classification, IEEE Trans. Inform. Theory 13 (1967) 21–27.

[11] T. Denoeux, A k-nearest neighbor classification rule based on dempster-shafer theory., IEEE Trans. Syst. Man. Cybern. 25 (1995) 804–813.

[12] S. Destercke, A Decision Rule for Imprecise Probabilities Based on Pair-Wise Comparison of Expectation Bounds, in: Combining Soft Computing and Statistical Methods in Data Analysis, Springer-Verlag Berlin, 2010, pp. 189–197.

[13] S. Destercke, A k-nearest neighbours method based on lower previsions, in: IPMU, 2010, pp. 129–138.

[14] D. Dubois, H. Prade, Possibility Theory: An Approach to Computerized Processing of Uncertainty, Plenum Press, New York, 1988.

[15] B. Dubuisson, M. Masson, A statistical decision rule with incomplete knowledge about classes, Pattern Recognition 26 (1993) 155–165.

[16] S. Dudani, The distance-weighted k-nearest neighbor rule, IEEE Trans. Syst. Man. Cybern. 6 (1976).

[17] B. Finetti, Theory of probability, volume 1-2, Wiley, NY, 1974. Translation of 1970 book.

[18] E. Fix, J. Hodges, Discriminatory analysis, nonparametric discrimination: consistency properties., Technical Report 4, USAF School of Aviation Medicine, 1951.

[19] I. Gilboa, D. Schmeidler, Maxmin expected utility with non-unique prior, Journal of Mathematical Economics 18 (1989) 141–153.

[20] T. Hastie, R. Tibshirani, Discriminant adaptive nearest neighbor classification, IEEE Trans. Pattern Anal. Mach. Intell. 18 (1996) 607–616.

[21] C.H. Hsieh, Y.J. Liu, Fast search algorithms for vector quantization of images using multiple triangle inequalities and wavelet transform, IEEE Trans. Image Process. 9 (2000) 321–328.

[22] E. Hüllermeier, Case-based approximate reasoning, volume 44 of *Theory and decision library*, Springer, 2007.

[23] J.Y. Jaffray, M. Jeleva, Information processing under imprecise risk with the Hurwicz criterion, in: Proc. of the fifth Int. Symp. on Imprecise Probabilities and Their Applications, 2007.

[24] J. Keller, M. Gray, J. Givens, A fuzzy k-nn neighbor algorithm, IEEE Trans. Syst. Man. Cybern. 15 (1985) 580–585.

[25] E. Miranda, A survey of the theory of coherent lower previsions, Int. J. of Approximate Reasoning 48 (2008) 628–658.

[26] S. Moral, J. Sagrado, Aggregation of imprecise probabilities, in: B. BouchonMeunier (Ed.), Aggregation and Fusion of Imperfect Information, Physica-Verlag, Heidelberg, 1997, pp. 162–188.

[27] E. Quaeghebeur, G.D. Cooman, Extreme lower probabilities, Fuzzy Sets and Systems 159 (2008) 2163–2175.

[28] G. Regoli, Inference under imprecise probability assessments, Soft Comput. 3 (1999) 181–186.

[29] G. Shafer, A mathematical Theory of Evidence, Princeton University Press, New Jersey, 1976.

[30] P. Smets, R. Kennes, The transferable belief model, Artificial Intelligence 66 (1994) 191–234.

[31] C. Tomasi, R. Manduchi, Stereo matching as a nearest-neighbor problem, IEEE Trans. Pattern Anal. Mach. Intell. 20 (1998) 333–340.

[32] J. Toyama, M. Kudo, H. Imai, Probably correct k-nearest neighbor search in high dimensions, Pattern Recognition 43 (2010) 1361–1372.

[33] M. Troffaes, Decision making under uncertainty using imprecise probabilities, Int. J. of Approximate Reasoning 45 (2007) 17–29.

[34] C.F. Tsai, C.Y. Lin, A triangle area based nearest neighbors approach to intrusion detection, Pattern Recognition 43 (2010) 222–229.

[35] G. Tsoumakas, I.P. Vlahavas, Random-labelsets: An ensemble method for multilabel classification, in: ECML, 2007, pp. 406–417.

[36] L. Utkin, Extensions of belief functions and possibility distributions by using the imprecise dirichlet model., Fuzzy Sets and Systems 154 (2005) 413–431.

[37] P. Walley, The elicitation and aggregation of beliefs, Technical Report, University of Warwick, 1982.

[38] P. Walley, Statistical reasoning with imprecise Probabilities, Chapman and Hall, New York, 1991.

[39] J. Wang, P. Neskovic, L. Cooper, Improving nearest neighbor rule with a simple adaptive distance measure, Pattern Recognition Letters 28 (2007) 207–213.

[40] P.M. Williams, Notes on conditional previsions, Int. J. Approx. Reasoning 44 (2007) 366–383.

[41] L. Zadeh, The concept of a linguistic variable and its application to approximate reasoning-i, Information Sciences 8 (1975) 199–249.

[42] M. Zaffalon, G. Corani, D. Maua, Utility-based accuracy measures to empirically evaluate credal classifiers, in: F. Coolen, G. de Cooman, T. Fetz, M. Oberguggenberger (Eds.), ISIPTA 11, 2011, pp. 401–410.

[43] C.Y. Zhou, Y.Q. Chen., Improving nearest neighbor classification with cam weighted distance, Pattern Recognition 39 (2006) 635–645.

[44] L. Zouhal, T. Denoeux, An evidence-theoretic k-nn rule with parameter optimization, IEEE Trans. on Syst., Man, and Cybern. 28 (1998) 263–271.