Nonparametric Regression Analysis of Uncertain and Imprecise Data using Belief Functions

Simon Petit-Renaud[†,*] and Thierry Denœux[†1]

[†]Université de Technologie de Compiègne

UMR CNRS 6599 Heudiasyc

Centre de Recherches de Royallieu

BP 20529 - F-60205 Compiègne cedex - France

tdenoeux@utc.fr

[*]Laboratoire d'Informatique de l'Université du Maine,

72085 Le Mans Cedex 9, France

simon.petit-renaud@lium.univ-lemans.fr

June 3, 2003

[1]Corresponding author.

Abstract

This paper introduces a new approach to regression analysis based on a fuzzy extension of belief function theory. For a given input vector $\mathbf{x}$, the method provides a prediction regarding the value of the output variable y, in the form of a fuzzy belief assignment (FBA), defined as a collection of fuzzy sets of values with associated masses of belief. The output FBA is computed using a nonparametric, instance-based approach: training samples in the neighborhood of $\mathbf{x}$ are considered as sources of partial information on the response variable; the pieces of evidence are discounted as a function of their distance to $\mathbf{x}$, and pooled using Dempster's rule of combination. The method can cope with heterogenous training data, including numbers, intervals, fuzzy numbers, and, more generally, fuzzy belief assignments, a convenient formalism for modeling unreliable and imprecise information provided by experts or multi-sensor systems. The performances of the method are compared to those of standard regression techniques using several simulated data sets.

Keywords: Dempster-Shafer Theory, Evidence Theory, Transferable Belief Model, Fuzzy data, Imprecise data, Uncertainty, Regression analysis, function approximation, supervised learning.

1 Introduction

Learning plays a central role in many fields such as Statistics, Artificial Intelligence, and Pattern Recognition [13]. In particular, *supervised learning* is concerned with the prediction of a response (or output) variable y, based on a vector $\mathbf{x} = (x_1, \ldots, x_d)$ of d observed input variables, or predictors. This problem is also referred to as *classification* when the output y is qualitative, and *regression* when it is a quantitative measurement.

Classically, the available information resides in a learning set $\mathcal{L} = \{(\mathbf{x}_i, y_i)\}_{i=1}^N$ of N observations of the input and output variables. It is customary to consider these observations as being drawn independently from a joint probability measure $F(\mathbf{x}, y)$. A general principle consists in determining, among all the measurable functions g, the function of the input $\mathbf{x}$ which best explains the output y, according to a given criterion, such as the mean squared error:

$$R(g) = \int (y - g(\mathbf{x}))^2 dF(\mathbf{x}, y). \tag{1}$$

According to this criterion, the best predictions are achieved by the regression function, defined as the conditional expectation of y given $\mathbf{x}$. Many techniques have been proposed in the literature to estimate the regression function, such as kernel or nearest-neighbor methods, smoothing splines, multi-layer perceptrons, radial basis function networks, projection pursuit methods, etc. (see, e.g. [13] for a recent overview of these methods). These techniques have proved very efficient in a wide range of situations. However, they suffer from certain limitations.

In particular, classical regression techniques assume perfect knowledge of the value of the response variable y for the learning examples. That is to say, the observations are supposed to be both precise (point-valued) and certain. There are, however, situations in which this assumption is not realistic. Quite often, information about y is obtained through measuring devices, or sensors, with limited precision and reliability. *Imprecise* observations of the responses may then be better modeled by real intervals $[y_i^-, y_i^+]$ or fuzzy numbers $\widetilde{y}_i$. Several approaches have been proposed for processing such learning data, such as interval or fuzzy linear regression [8], and fuzzy [25, 28] or neuro-fuzzy inference systems [15]. However, the *uncertainty* of observations due, e.g., to poor sensor reliability, is not easily taken into account in these approaches.

An observation may be imprecise, uncertain, or both, and these situations must be properly represented in a learning system (see [9][22] for discussions concerning the notions of imprecision and uncertainty). Occasionally, the situation is even more complex, and the quantity y of interest is observed by several sensors, with different degrees of accuracy. We then need a formalism to handle such imprecise and partially conflicting data.

Additionally, a learning system processing such information should reflect in its outputs not only the quality of the training data, but also the relevance of this data to the current prediction task at hand. In particular, if the current input vector $\mathbf{x}$ is very dissimilar from all training input vectors $\mathbf{x}_i$, some doubt should be cast on the validity on the prediction, and this should be reflected in the system output. This property is rarely verified in conventional statistical methods, which are essentially based on asymptotic assumptions (the learning set is assumed to be large enough to cover the whole observation space).

To address the above issues, we propose a new approach to regression analysis based on a fuzzy extension of belief function theory (also called Dempster-Shafer, or Evidence theory). Whereas a basic belief assignment in "standard" or "crisp" evidence theory assigns belief masses to crisp subsets of the possibility space (or "frame of discernment") Ω, a fuzzy belief assignment (FBA) allocates parts of a unit mass of belief to fuzzy subsets of Ω. The concept of FBA thus subsumes those of crisp and fuzzy sets, as well as crisp belief assignments. Our FBA-based regression method, called EVREG (Evidential Regression) generalizes an instance-based approach introduced in classification by one of the authors [2, 31, 5, 7]. Basically, the method considers each training sample in the neighborhood of the input vector $\mathbf{x}$ as a piece of evidence regarding the value of the output y. The pieces of evidence are discounted as a function of their distance to $\mathbf{x}$, and pooled using Dempster's rule of combination. The result is a FBA that quantifies one's beliefs concerning the value of y, based on the learning set information. A probability density function and a point prediction can be computed from this FBA, providing the user with information at several levels of detail.

This article is organized as follows. The background on belief function theory and its fuzzy extension is first recalled in Section 2. The EVREG method is then introduced in Section 3, and a parameter optimization procedure is described in Section 4. Finally, experimental results are presented in Section 5, and Section 6 concludes the paper.

2 Fuzzy Evidence Theory

2.1 Belief Function Theory

In this section, we briefly introduce the necessary notions of belief function theory [1, 19, 24]. The interested reader is referred to, e.g., [19, 24, 20] for mathematical developments and in-depth discussion on possible interpretations of the theory. In this paper, we shall adopt the subjectivist, non probabilistic view of Smets' Transferable Belief Model (TBM) [24, 20]. The aim of this model is to represent the belief of an agent concerning the value of a given variable y, based on available information, and to propose rules whereby the agent's beliefs can be updated when new evidence is gathered.

Let Ω be a finite set, and let 2^Ω be the set of all subsets of Ω. The fundamental concept for representing uncertainty about y, given an evidential corpus EC, is that of basic belief assignment (BBA), also called belief structure or mass function, defined as a function $m_y[EC]$ from 2^Ω to $[0, 1]$ verifying:

$$\sum_{A \subseteq \Omega} m_y[EC](A) = 1.$$

The quantity $m_y[EC](A)$ represents the belief alloted to the proposition $y \in A$, and that cannot be assigned to any more restrictive proposition, given the available knowledge. When the context makes clear what the reference variable and the evidential corpus are, the notation m_y, or even m will be used instead of $m_y[EC]$. A BBA m such that $m(\emptyset) = 0$ is said to be *normal* (this condition is not imposed in the TBM). Any subset of Ω such as $m(A) > 0$ is called a focal element of m. We will denote by $\mathcal{F}(m)$ the set of focal elements of m. The information provided by a BBA can be represented by a *belief function* or by a *plausibility function* defined, respectively, as:

$$\mathrm{bel}(A) = \sum_{\emptyset \neq B \subseteq A} m(B)$$

and

$$\mathrm{pl}(A) = \sum_{B \cap A \neq \emptyset} m(B) = \mathrm{bel}(\Omega) - \mathrm{bel}(\overline{A}).$$

The quantity $\mathrm{bel}(A)$ is interpreted as the *total* belief committed to A and $\mathrm{pl}(A)$ as the belief that *might* be committed to A, if further information became available. One can show that the three functions, m, bel and pl, are in one-to-one correspondence and that bel and pl are monotonous functions of infinite order [19]. Belief and plausibility measures boil down to probability measures in the special case where all the focal elements are singletons of Ω. Another special case is the vacuous BBA verifying $m(\Omega) = 1$. This represents complete ignorance regarding the value of y.

One of the most important operations in the theory is the procedure for aggregating multiple BBA's on the same variable. Let us suppose that two distinct sources separately induce two BBA's m_1 and m_2. For any binary set operation ∇, the fusion of these BA's, noted $m = m_1 \circledtriangledown m_2$, may be defined as [28]:

$$m(A) = \sum_{B \nabla C = A} m_1(B) m_2(C), \quad \forall A \in \Omega. \tag{2}$$

The conjunctive rule is obtained by choosing $\nabla = \cap$, and the disjunctive rule by setting $\nabla = \cup$. Note that the conjunctive rule $\textcircled{\cap}$ may produce a subnormal BA, i.e. one may have $m(\emptyset) > 0$. The Dempster normalization procedure converts a subnormal BBA m into a normal one m^* defined as follows:

$$m^*(A) = \frac{m(A)}{1 - m(\emptyset)} \tag{3}$$

for $A \neq \emptyset$ and $m^*(\emptyset) = 0$. The so-called Dempster's rule of combination [1, 19], noted $\oplus$ corresponds to the conjunctive sum followed by Dempster's normalization. The conjunctive and Dempster's rules of combination are relevant when all the sources to be combined are distinct and reliable. These operations are associative, commutative and have the vacuous BBA as neutral element.

It sometimes occurs that a source of information induces a bba m, but we have some doubt regarding the reliability of that source. Such metaknowledge may be represented by discounting [19] m by some factor $\alpha \in [0, 1]$, which leads to a bba m^α

defined as:

$$m^{\alpha}(A) = (1-\alpha)m(A) \qquad \forall\, A \subseteq \Omega, A \neq \Omega \tag{4}$$

$$m^{\alpha}(\Omega) = \alpha + (1-\alpha)m(\Omega) \tag{5}$$

A discount rate $\alpha = 1$ means that one is sure that the source cannot be trusted: the resulting BBA is then vacuous. On the contrary, a null discount rate leaves m unchanged: this corresponds to the situation in which the source is known to be fully reliable.

Any one of the functions m, bel and pl describes a belief state. In the TBM, this "credal level" is distinct from the "decision level" where decision making takes place [23]. As remarked by Smets [23], the use of probabilities in a decision context is strongly supported by rationality arguments. A belief function thus has to be transformed into a probability function for decision making. The only transformation satisfying certain axiomatic requirements was shown by Smets to be the pignistic transformation [23], in which each mass of belief $m(A)$ is distributed equally among the elements of A for all $A \subseteq \Omega$. This leads to the pignistic probability distribution defined as:

$$p_{bet}(\omega) = \sum_{\{A \subseteq \Omega, A \neq \emptyset\}} m^*(A) \frac{A(\omega)}{|A|} \tag{6}$$

where $A(\cdot)$ denotes the characteristic function of A and $|A|$ its cardinality.

BF theory can easily be generalized to continuous spaces provided the number of focal elements $|\mathcal{F}(m)|$ remains finite[1]. In this case, all the expressions defined above are unchanged, except that the cardinality of a given set A is replaced by its Lebesgue measure:

$$|A| = \int A(\omega) d\omega.$$

When $|A| < \infty$ for all $A \in \mathcal{F}(m)$, the pignistic probability function still exists but becomes a probability density function. In particular, if $\Omega \subset \mathbb{R}$, and the focal elements A of m are bounded intervals, p_{bet} is a finite mixture of continuous uniform distributions.

[1] A more general extension is possible, requiring more complex measure-theoretic concepts. This extension will not be considered in this paper.

2.2 Fuzzy Extension

The above formalism can be generalized in order to represent belief in fuzzy propositions, such as "y is high". Fuzzy extensions of evidence theory have been proposed by different authors [30, 21, 26, 29, 4]. The basic idea is to allow the focal elements of a BBA to be fuzzy sets. If A is a fuzzy subset of Ω, we will denote by $A(\cdot)$ its membership function and by $h(A)$ its height. Let $[0,1]^\Omega$ denote the set of fuzzy sets of Ω. A fuzzy belief assignment (FBA), also called a fuzzy belief structure, is a function m from $[0,1]^\Omega$ to $[0,1]$ such that, for some finite collection $\mathcal{F}(m)$ of fuzzy subsets of Ω,

$$m(A) > 0 \quad \forall A \in \mathcal{F}(m) \tag{7}$$

$$m(A) = 0 \quad \forall A \notin \mathcal{F}(m) \tag{8}$$

$$\sum_{A \in \mathcal{F}(m)} m(A) = 1 \ . \tag{9}$$

Here again, the elements of $\mathcal{F}(m)$ are called the focal elements of m. If all the focal elements are normalized (i.e. $h(A) = 1$ for each $A \in \mathcal{F}(m)$), m is said to be normal. Yager [27] proposed a "smooth normalization procedure" (SNP) for converting a subnormal FBA into a normal one. This method generalizes both fuzzy set normalization and Dempster's normalization of crisp BA's (3). It is defined as:

$$m^*(A) = \frac{\sum_{B^*=A} h(B)m(B)}{\sum_{C \in \mathcal{F}(m)} h(C)m(C)} \tag{10}$$

where B^* is the normal fuzzy set defined by $B^*(\omega) = B(\omega)/h(B)$, $\forall \omega \in \Omega$.

Since the belief and plausibility functions are based on set-theoretic operations (inclusion and intersection), their expression can be generalized as follows [26]:

$$\mathrm{pl}(A) = \sum_{B \in \mathcal{F}(m)} m(B)\mathrm{Int}(A, B), \tag{11}$$

$$\mathrm{bel}(A) = \sum_{B \in \mathcal{F}(m)} m(B)\mathrm{Inc}_A(B), \tag{12}$$

where $\mathrm{Int}(A, B)$ and $\mathrm{Inc}_A(B)$ are, respectively, a measure of intersection between A and B, and an inclusion measure of B in A. Using the standard fuzzy union and intersection operators, these measures can be defined as follows:

$$\mathrm{Int}(A, B) = \sup_{\omega \in \Omega} \min\left(A(\omega), B(\omega)\right)$$

$$\mathrm{Inc}_A(B) = \inf_{\omega \in \Omega} \max\left(A(\omega), (1 - B(\omega)\right).$$

Combination operations can also be generalized, by using an appropriate fuzzy set operator in (2). In particular, a fuzzy version of the conjunctive sum is obtained by using the standard fuzzy intersection.

Finally, the pignistic probability function may be defined as:

$$p_{bet}(\omega) = \sum_{A \in \mathcal{F}(m^*)} \frac{m^*(A)}{|A|} A(\omega), \quad \forall \omega \in \Omega, \tag{13}$$

where $|A|$ is the sigma-count cardinality of A:

$$|A| = \begin{cases} \sum_{\omega \in \Omega} A(\omega) & \text{if } \Omega \text{ is finite} \\ \int A(\omega) d\omega & \text{if } \Omega \text{ is continuous.} \end{cases}$$

3 Application to Regression

3.1 The Data

In this section, we show how to use the above concepts of fuzzy evidence theory in the regression analysis framework [16]. This approach extends that introduced by Denœux [2, 5] in the context of supervised classification.

We assume the training data to be of the form:

$$\mathcal{L} = \{e_i = (\mathbf{x}_i, m_i)\}_{i=1}^{N} \tag{14}$$

where $\mathbf{x}_i$ is the input vector for example e_i, and m_i is a FBA on an ordered or continuous frame $\mathcal{Y}$, which quantifies one's partial knowledge of the value taken by the response variable y_i for example e_i (a more rigorous, but cumbersome notation would be $m_{y_i}[EC]$, where EC is the evidential corpus on which the available knowledge on y_i is based). Using this very general formalism, it is possible to model various types of training data. In particular, a classical learning set is recovered when all BA's m_i are focused on a unique singleton y_i. The interval and fuzzy regression situations correspond to the case where each m_i has, respectively, an interval or a fuzzy number as a unique focal element. The most general situation is that of general FBA's, with focal elements $\mathcal{F}(m_i) = \{F_{ij}\}_{j=1}^{J(i)}$, where the F_{ij} are fuzzy subsets of $\mathbb{R}$, and $J(i)$ is the number of focal elements of m_i.

3.2 The EVREG model

Let $\mathbf{x}$ be an arbitrary vector, and y the corresponding unknown output. The problem is now to deduce some information on y from the training set $\mathcal{L}$. Since the learning set information is potentially imprecise and uncertain, the output will take the form of a FBA on $\mathcal{Y}$ denoted $m_y[\mathbf{x}, \mathcal{L}]$. As proposed in [2, 5] in the context of classification, this FBA can be constructed in two steps: *discounting* of the FBA's m_i, $i = 1, \ldots, N$ according to a measure of dissimilarity between input vectors, and combination of the discounted FBA's.

Each element $e_i = (\mathbf{x}_i, m_i)$ of the training set is a piece of evidence concerning the possible value of y_i, which can be represented by a FBA $m_y[\mathbf{x}, e_i]$. The relevance of that information regarding the variable of interest y can reasonably be assumed to depend on the *dissimilarity*, measured by a suitable distance function, between input vectors $\mathbf{x}$ and $\mathbf{x}_i$. If $\mathbf{x}$ is "close" to $\mathbf{x}_i$ according to a given metric $\|.\|$, y can be expected to be close to y_i, which makes example e_i quite relevant to predict the value of y. On the contrary, if $\mathbf{x}$ and $\mathbf{x}_i$ are very dissimilar, example e_i provides only marginal information regarding the value of y. More formally, we propose to define $m_y[\mathbf{x}, e_i]$ as a *discounting* of m_i:

$$m_y[\mathbf{x}, e_i](A) = \begin{cases} m_i(A)\phi(\|\mathbf{x} - \mathbf{x}_i\|) & \text{if } A \in \mathcal{F}(m_i) \setminus \{\mathcal{Y}\} \\ 1 - \phi(\|\mathbf{x} - \mathbf{x}_i\|) & \text{if } A = \mathcal{Y} \\ 0 & \text{otherwise,} \end{cases} \tag{15}$$

where ϕ is a decreasing function from $\mathbb{R}^+$ to $[0,1]$ verifying $\phi(0) \in]0,1[$ and

$$\lim_{d \to \infty} \phi(d) = 0. \tag{16}$$

In (15), the discount rate $\alpha_i = 1 - \phi(\|\mathbf{x} - \mathbf{x}_i\|)$ determines the influence of $\mathbf{x}_i$ on $\mathbf{x}$. If $\mathbf{x}$ is close to $\mathbf{x}_i$, α_i is close to 0 and the mass functions $m_y[\mathbf{x}, e_i]$ and m_i are very similar. When $\mathbf{x}$ is very far from $\mathbf{x}_i$, α_i tends to 1 and $m_y[\mathbf{x}, e_i]$ tends to the vacuous belief assignment ($m_y[\mathbf{x}, e_i](\mathcal{Y}) \approx 1$). In the following, function ϕ will be referred to as a *discounting function*. As shown in [3], when the metric is defined as:

$$\|\mathbf{x} - \mathbf{x}_i\| = \left[(\mathbf{x} - \mathbf{x}_i)^T \boldsymbol{\Sigma}^{-1} (\mathbf{x} - \mathbf{x}_i)\right]^{1/2}, \tag{17}$$

where $\mathbf{\Sigma}$ is a symmetric positive definite matrix, a natural choice for ϕ is:

$$\phi(d) = \gamma \exp(-d^2), \tag{18}$$

where $\gamma \in]0, 1]$ is a real parameter.

In order to combine the information provided by each element of the training set, we can use the conjunctive rule of combination for FBA's. The choice of a conjunctive operator is justified by the neutral property of the vacuous BA: it is essential that a vector $\mathbf{x}_i$ which is far from $\mathbf{x}$ has very little influence on the estimation of the corresponding y. The final BA is then:

$$m_y[\mathbf{x}, \mathcal{L}] = \textcircled{\cap}_{i=1}^{N} m_y[\mathbf{x}, e_i]. \tag{19}$$

We denote by $m_y^*[\mathbf{x}, \mathcal{L}]$ the FBA obtained by normalizing $m_y[\mathbf{x}, \mathcal{L}]$ using the SNP (10).

Example 1 Let us consider as an example an output BA $m_y[\mathbf{x}, e_1, e_2]$ computed from two elements $e_1 = (\mathbf{x}_1, m_1)$ and $e_2 = (\mathbf{x}_2, m_2)$ of a training set. The frame of discernment $\mathcal{Y}$ is the interval $[0, 10]$. The belief assignments m_1 and m_2 are focused on triangular fuzzy numbers (Figure 1). A triangular fuzzy number with support $[a, c]$ and core b will be noted $(a, b, c)_T$. Let $A = (0, 2, 4)_T$, $B = (2, 4, 6)_T$, and $C = (3, 6, 8)_T$ be three triangular fuzzy numbers, and let m_1 and m_2 be defined as:

$$m_1(A) = 0.4 \qquad m_1(B) = 0.6$$

$$m_2(C) = 1.$$

We assume that vector $\mathbf{x}$ is closer to $\mathbf{x}_1$ than to $\mathbf{x}_2$ and, more precisely, that the discount rates are $\alpha_1 = 0.2$ and $\alpha_2 = 0.6$. Consequently, the influence of $\mathbf{x}_2$ will be weaker than that of $\mathbf{x}_1$. The calculation of $m_y[\mathbf{x}, e_1, e_2]$ is divided in 2 steps:

1. Discounting of m_i, $i = 1, 2$: We have

$$m_y[\mathbf{x}, e_1](A) = 0.4 \times (1 - 0.2) = 0.32,$$

$$m_y[\mathbf{x}, e_1](B) = 0.6 \times (1 - 0.2) = 0.48,$$

$$m_y[\mathbf{x}, e_1](\mathcal{Y}) = 1 - 0.32 - 0.48 = 0.2,$$

and

$$m_y[\mathbf{x}, e_2](C) = 1 \times (1 - 0.6) = 0.4,$$

$$m_y[\mathbf{x}, e_2](\mathcal{Y}) = 1 - 0.4 = 0.6$$

2. Combination:

$$m_y[\mathbf{x}, e_1, e_2](A) = 0.192 \qquad m_y[\mathbf{x}, e_1, e_2](B) = 0.288,$$

$$m_y[\mathbf{x}, e_1, e_2](C) = 0.08 \qquad m_y[\mathbf{x}, e_1, e_2](\mathcal{Y}) = 0.12,$$

$$m_y[\mathbf{x}, e_1, e_2](A \cap C) = 0.128 \qquad m_y[\mathbf{x}, e_1, e_2](B \cap C) = 0.192.$$

Remark 1 The number of focal elements of $m_y[\mathbf{x}, \mathcal{L}]$ can, in the worst case, increase exponentially with the number of BA's combined, making the computation very heavy for large N. A simple but efficient way to avoid this problem is to compute the belief assignments provided only by the k nearest neighbors $\{\mathbf{x}_{(i)}\}_{i=1}^{k}$ of $\mathbf{x}$ in the training set:

$$m_y[\mathbf{x}, \mathcal{L}] = \circledcap_{i=1}^{k} m_y[\mathbf{x}, e_{(i)}] \tag{20}$$

An additional way to speed up the computations is to simplify BA's by aggregating similar, or unimportant focal elements, thus reducing the number of focal elements to take into account in the combination. This method has been introduced for regression problems in [17, 18], and extended in [6].

Remark 2 In the case of a classical training set, each BBA m_i has a single focal element $\{y_i\}$. Consequently, $m_y[\mathbf{x}, \mathcal{L}]$ is then a crisp BBA with $N + 1$ focal elements. Its normalized version has the following expression (assuming the y_i to be all different) as:

$$m_y^*[\mathbf{x}, \mathcal{L}](A) = \begin{cases} \frac{1}{K}\phi(\|\mathbf{x} - \mathbf{x}_i\|) \prod_{j \neq i}(1 - \phi(\|\mathbf{x} - \mathbf{x}_j\|)) & \text{if } A = y_i, \\ \frac{1}{K} \prod_{i=1}^{N} (1 - \phi(\|\mathbf{x} - \mathbf{x}_i\|)) & \text{if } A = \mathcal{Y} \\ 0 & \text{otherwise,} \end{cases}$$

where

$$K = \prod_{i=1}^{N}(1 - \phi(\|\mathbf{x} - \mathbf{x}_i\|) + \sum_{i=1}^{N} \phi(\|\mathbf{x} - \mathbf{x}_i\|) \prod_{j \neq i}(1 - \phi(\|\mathbf{x} - \mathbf{x}_j\|)$$

is the normalization factor. Note that all strict subsets of $\mathcal{Y}$ that do not contain any of the y_i's do not receive any belief. This may seem somewhat paradoxical, as an observation y_i or the response variable in a neighborhood of $\mathbf{x}$ may be argued to make values close to y_i more likely at $\mathbf{x}$, provided the underlying input-output function is assumed to be continuous. In fact, the BBA $m_y[\mathbf{x}, \mathcal{L}]$ merely encodes the available evidence, without introducing any additional assumption (not even continuity). As we shall see in the sequel, interpolation is performed at a later stage, when computing the pignitic expectation of y given $\mathbf{x}$, where our method bares some resemblance with classical nonparametric smoothing techniques.

3.3 Point and Interval Prediction

3.3.1 Pignistic Expectation and Quantiles

Assuming the domain $\mathcal{Y}$ of y to be a bounded interval $[y_{inf}, y_{sup}]$, the probabilistic density function $p_{bet}[\mathbf{x}, \mathcal{L}]$ associated to $m[\mathbf{x}, \mathcal{L}]$ exists. It is defined by (6) for the crisp case, and by (13) for the fuzzy case, and has the following expression:

$$p_{bet}[\mathbf{x}, \mathcal{L}](y) = \sum_{A \in \mathcal{F}(m_y^*[\mathbf{x}, \mathcal{L}])} m_y^*[\mathbf{x}, \mathcal{L}](A) \frac{A(y)}{|A|}. \tag{21}$$

It is therefore a mixture of the probability densities defined by the normalized membership functions of the focal elements of $m_y^*[\mathbf{x}, \mathcal{L}]$. Note that, when $\mathbf{x}$ is very dissimilar from all the $\mathbf{x}_i$, $m_y^*[\mathbf{x}, \mathcal{L}]$ is close to the vacuous BA. In that case, $p_{bet}[\mathbf{x}, \mathcal{L}]$ is close to the uniform probability distribution on $\mathcal{Y}$.

The pignistic probability function allows to define some summary statistics, such as the median, quantiles, or the expectation defined as:

$$\widehat{y}(\mathbf{x}) = \sum_{A \in \mathcal{F}(m_y^*[\mathbf{x}, \mathcal{L}])} \frac{m_y^*[\mathbf{x}, \mathcal{L}](A)}{|A|} \int_{\mathcal{Y}} uA(u)du.$$

Let y_A^* be the center of gravity of A:

$$y_A^* = \frac{\int_{\mathcal{Y}} uA(u)du}{\int_{\mathcal{Y}} A(u)du}.$$

Then $\widehat{y}$ can be expressed as:

$$\widehat{y}(\mathbf{x}) = \sum_{A \in \mathcal{F}(m_y^*[\mathbf{x}, \mathcal{L}])} m_y^*[\mathbf{x}, \mathcal{L}](A)\, y_A^*. \tag{22}$$

Remark 3 In the particular case where the outputs are real numbers, the pignistic probability becomes:

$$p_{bet}[\mathbf{x},\mathcal{L}](y) = \sum_{i=1}^{N} m_y^*[\mathbf{x},\mathcal{L}](\{y_i\})\delta_{\{y_i\}}(y) + \frac{m_y^*[\mathbf{x},\mathcal{L}](\mathcal{Y})}{y_{sup} - y_{inf}}. \quad (23)$$

It is a mixture of Dirac distributions and a continuous uniform distribution. The expectation of $p_{bet}[\mathbf{x},\mathcal{L}]$ is then:

$$\widehat{y}(\mathbf{x}) = \sum_{i=1}^{N} m_y^*[\mathbf{x},\mathcal{L}](\{y_i\})y_i + m_y^*[\mathbf{x},\mathcal{L}](\mathcal{Y})\overline{y}, \quad (24)$$

with $\overline{y} = (y_{inf} + y_{sup})/2$. Considering $\overline{y}$ as an additional observation y_{N+1}, the resulting regression function is then a linear function of the y_i and can be written:

$$\widehat{y}(\mathbf{x}) = \sum_{i=1}^{N+1} S_i y_i \quad (25)$$

where S_i is a weight depending on $\mathbf{x}$ and all the $\mathbf{x}_j$, $j = 1, \ldots, N$. The sequence of weights $(S_i)_{i=1}^{N+1}$ defines the *equivalent kernel* at $\mathbf{x}$ [14, page 20]. In the case of "classical" training data, our method can therefore by compared on common grounds with other nonparametric regression techniques such as spline or kernel smoothers.

3.3.2 Upper and Lower Expectations

Let us first assume that the BA's are crisp. Beside the pignistic expectation, other definitions of expectation have been proposed for belief functions. In particular, in the crisp case, the lower and upper expectations are defined, respectively, as follows [21]:

$$\widehat{y}^*(\mathbf{x}) = \sum_{A\in\mathcal{F}(m_y^*[\mathbf{x},\mathcal{L}])} m_y^*[\mathbf{x},\mathcal{L}](A) \sup_{y\in A} y, \quad (26)$$

$$\widehat{y}_*(\mathbf{x}) = \sum_{A\in\mathcal{F}(m_y^*[\mathbf{x},\mathcal{L}])} m_y^*[\mathbf{x},\mathcal{L}](A) \inf_{y\in A} y. \quad (27)$$

We observe that the interval $[\widehat{y}_*(\mathbf{x}), \widehat{y}^*(\mathbf{x})]$ contains the pignistic expectation $\widehat{y}(\mathbf{x})$. Its width may be interpreted as reflecting the uncertainty of the prediction.

In the particular case where the outputs are real numbers, the upper and lower expectation become:

$$\widehat{y}^*(\mathbf{x}) = \sum_{i=1}^{N} m_y^*[\mathbf{x},\mathcal{L}](\{y_i\})y_i + m_y^*[\mathbf{x},\mathcal{L}](\mathcal{Y})y_{sup}, \quad (28)$$

$$\widehat{y}_*(\mathbf{x}) = \sum_{i=1}^{N} m_y^*[\mathbf{x}, \mathcal{L}](\{y_i\}) y_i + m_y^*[\mathbf{x}, \mathcal{L}](\mathcal{Y}) y_{inf}. \tag{29}$$

If the belief assignments are fuzzy and their focal elements are fuzzy numbers, Dubois and Prade [10] have proposed to generalize the lower-upper expectation interval as the fuzzy number:

$$\widetilde{y}(\mathbf{x}) = \sum_{\widetilde{A} \in \mathcal{F}(m_y^*[\mathbf{x}, \mathcal{L}])} m_y^*[\mathbf{x}, \mathcal{L}](A) \cdot \widetilde{A} \tag{30}$$

where $\sum$ denotes the addition of fuzzy numbers [9]. A more general approach can be based on the decomposition of fuzzy focal elements A in α-cuts A^α. For each $\alpha \in [0, 1]$, each α-cut of the lower expectation is computed as in (27):

$$\widehat{y}_*^\alpha(\mathbf{x}) = \sum_{A \in \mathcal{F}(m_y[\mathbf{x}, \mathcal{L}])} m[\mathbf{x}, \mathcal{L}](A) \inf_{y \in A_\alpha} y. \tag{31}$$

The upper expectation is obtained in the same manner.

4 Learning

4.1 Performance Assessment

In Section 3, we have seen that the proposed model depends on a discounting function ϕ and a dissimilarity measure $\|.\|$. The choice of ϕ and $\|.\|$ is important for optimizing the performances of the method. To simplify this problem, we suppose that these functions are chosen among a set Φ of functions ϕ_θ indexed by a scalar or vector parameter $\theta \in \Theta$. For example, if $\|.\|$ and ϕ are defined using (17) and (18), then $\theta = (\gamma, \mathbf{\Sigma})$.

Once the set Φ of functions and an error criterion have been defined, it is possible to optimize parameter θ. However, we have only assumed partial knowledge of the response variable y_i for each learning example. Consequently, we need to define an error criterion allowing to compare an output FBA $m_y[\mathbf{x}, \mathcal{L}]$ with partial training information also represented by a FBA m. In the case of numerical output $\widehat{y}$ and a desired value y, a classical criterion is the squared error $(\widehat{y} - y)^2$. This criterion may be extended in several ways, starting from fuzzy sets of the real line, and finally FBA's.

Among the numerous distance measures between fuzzy sets defined in the literature [32], we propose to use a generalization of the quadratic version of the Hausdorff measure. In order to extend the mean squared error criterion, we slightly modify the Hausdorff distance as follows. Let $I_1 = [I_1^-, I_1^+]$ and $I_2 = [I_2^-, I_2^+]$ be two real intervals. The distance between I_1 and I_2 can be defined as:

$$h_2(I_1, I_2) = \max\{(I_1^- - I_2^-)^2, (I_1^+ - I_2^+)^2\}.$$

The distance between two *fuzzy intervals* F_1 and F_2 can then be defined as:

$$\widetilde{d}(F_1, F_2) = \int_0^1 h_2(F_1^\alpha, F_2^\alpha) d\alpha, \tag{32}$$

where F^α is the α-cut of F. If F_1 and F_2 are normal, but not necessarily convex, we can use (32), with

$$h_2(F_1^\alpha, F_2^\alpha) = \max\{(F_1^{\alpha -} - F_2^{\alpha -})^2, (F_1^{\alpha +} - F_2^{\alpha +})^2\}, \tag{33}$$

where $F^{\alpha -} = \inf_{y \in \mathcal{Y}} F^\alpha(y)$ and $F^{\alpha +} = \sup_{y \in \mathcal{Y}} F^\alpha(y)$.

Finally, we define the error between two normal FBA's m_1 and m_2 as:

$$C(m_1, m_2) = \sum_{F_1 \in \mathcal{F}(m_1)} \sum_{F_2 \in \mathcal{F}(m_2)} m_1(F_1) m_2(F_2) \widetilde{d}(F_1, F_2). \tag{34}$$

4.2 Parameter Estimation

Once a performance measure has been defined, classical model selection methods can be used for optimizing parameter θ, including re-sampling techniques such as cross-validation, jackknife, bootstrap and their variants [13]. In the following, we use a well known version of cross-validation: the *leave-one-out* method.

In this method, the response for each vector $\mathbf{x}_i$ of the training set is estimated with the $N-1$ other examples $(\mathbf{x}_j, m_j), j \neq i$, by applying (15) and (19) to $\mathbf{x}_i$. Let $\mathcal{L}^{-i}$ denote the learning set without example i. For each $\theta \in \Theta$, we obtain a BA concerning y_i, based on $\mathbf{x}_i$ and $\mathcal{L}^{-i}$, denoted by: $m_{y_i}[\mathbf{x}_i, \mathcal{L}^{-i}, \theta]$. The discrepancy with the true mass function m_i can be measured using criterion C defined in (34). The global selection criterion CV is then defined as the mean value in the training set:

$$CV(\theta) = \frac{1}{N} \sum_{i=1}^{N} C(m_i, m^*_{y_i}[\mathbf{x}_i, \mathcal{L}^{-i}, \theta]). \tag{35}$$

The estimator $\widehat{\theta}$ of parameter θ is then obtained by minimizing this criterion:

$$\widehat{\theta} = \arg\min_{\theta} CV(\theta). \tag{36}$$

5 Experiments

5.1 Motorcycle Data

In this classical regression problem based on a real data set, the scalar input x represents the time (in milliseconds) after a simulated impact of a motorcycle against an obstacle. The response variable y is the head acceleration of a post mortem human test object (in g). This is a classical data set composed of 133 examples of the form $(x_i, y_i) \in \mathbb{R}^2$. The data set was split into two parts: 66 examples for training and 67 examples for the test. The prediction for each example was computed using (19), without k-nearest neighbor approximation. Parameter γ in (18) was set at 0.9, and parameter σ in the expression of the distance:

$$d^2(x, x_i) = \frac{(x_i - x)^2}{\sigma^2}$$

was optimized using the cross-validation procedure described in Section 4.2, yielding $\sigma = 4.21$. The domain of the response variable was defined as $\mathcal{Y} = [-150, 80]$.

Figure 2 shows the 0.1, 0.25, 0.5, 0.75, and 0.9 quantiles of the output pignistic distribution of $p_{bet}[x, \mathcal{L}]$ (up), as well as the upper, lower and pignistic expectations defined, respectively, by (26), (27) and (22) (down), as a function of x. The pignistic probability distribution for each input value x can be seen to reflect the uncertainty on the corresponding value of the output variable, taking into account both the scatter and the density of training data (the uncertainty is maximal in the $[30, 40]$ range in which the output values have high variability, and beyond 60 where no data is available). In contrast, the width of the lower-upper interval seems to reflect only the scarcity of training data, since it is essentially related, in the case of precise training data, to the mass $m[x, \mathcal{L}](\mathcal{Y})$ given to the whole domain of y. In particular, the lower, upper and pignistic expectations are very close to each other in regions of high density, because the mass assigned to $\mathcal{Y}$ is nearly negligible in this area; our approach then behaves as a classical one.

In order to ensure that EVREG performs reasonably well on such a classical task, we compared it to the Nadaraya-Watson (NW) and k-nearest neighbor (k-NN) smoothers [12]. These two methods were chosen because they appear to be the most similar to our approch in the conventional non parametric statistical framework. Given a classical learning set $\mathcal{L} = \{(x_i, y_i)\}_{i=1}^n$, and a continuous, bounded and symmetric real function K called a kernel, the NW estimate of the response y for input x is defined as:

$$\widehat{f}_{NW}(x) = \frac{\sum_{i=1}^n K_\lambda(x - y_i)y_i}{\sum_{i=1}^n K_\lambda(x - y_i)},$$

where $K_\lambda(u) = \lambda^{-1}K(u/\lambda)$ is the kernel scaled with bandwidth λ. In our simulations, we used a Gaussian kernel given by $K_\lambda(u) = \lambda^{-1}\exp(-u^2/\lambda)$.

The kNN estimate of y at x is defined as:

$$\widehat{f}_k(x) = \frac{1}{k}\sum_{i=1}^n W_{ki}(x)y_i,$$

where $\{W_{ki}(x)\}_{i=1}^n$ is a sequence of weights defined by:

$$W_{ki}(x) = \begin{cases} 1 & \text{if } x_i \text{ is one of the } k \text{ nearest observations of } x, \\ 0 & \text{otherwise.} \end{cases}$$

The bandwidth λ in the NW method and the number k of neighbors in the k-NN regression method were determined using the training data by leave-one-out cross-validation, yielding $\lambda = 0.8$ and $k = 7$. The test mean squared error was 394 using the k-NN method, 467 using the NW method, and 433 using EVREG (Figure 3). The three models thus appear to be roughly equivalent in terms of prediction accuracy on such a classical task.

5.2 Unreliable Sensor

Problem Description and Data Generation

In this second example, we consider the situation in which the value of the response variable y is given by a sensor, the accuracy and reliability of which varies over time. Each learning example is thus assumed to be of the form $e_i = (x_i, z_i, \sigma_i, p_i)$, where x_i is the known input value, z_i is the measurement of the true (unknown) output y_i, σ_i is the standard deviation of the measurement error, and p_i is the probability that the sensor is in good operating condition. Therefore, σ_i characterizes the accuracy of the sensor (with a lower σ_i corresponding to a more accurate sensor), whereas p_i characterizes the sensor reliability (a higher value of p_i indicating a more reliable measurement value). Data of this kind may be encountered in situations where the accuracy and reliability of sensors vary with time (e.g., as a function of the time since the last maintenance operation), or depend on the context of the measurement, which frequently occurs, for instance, in remote sensing and target tracking applications (see, e.g., [11] for more discussion on this topic).

Data sets of the form above were generated using the following procedure. First, $N = 21$ input values were sampled regularly in the interval $\mathcal{Y} = [0, 10]$:

$$x_i = 0.5(i-1), \quad i = 1, \ldots, N. \tag{37}$$

The true output values were then computed as a deterministic function of the inputs:

$$y_i = x_i \sin x_i \quad i = 1, \ldots, N. \tag{38}$$

In real applications, the accuracy and reliability of a sensor may depend on several context factors such as the environment, the time since the last maintenance operation, etc. To simulate this variability, parameters σ_i and p_i were generated randomly from uniform distributions:

$$\sigma_i \sim \mathcal{U}_{[0.2,2.2]} \quad i = 1, \ldots, N. \tag{39}$$

$$p_i \sim \mathcal{U}_{[0,1]} \quad i = 1, \ldots, N. \tag{40}$$

Next, the state s_i of the sensor for each example i was simulated. The sensor was assumed to be in good operating condition ($s_i = 1$) with probability p_i, and to be broken ($s_i = 0$) with probability $1 - p_i$. Thus, s_i has a Bernouilli distribution $\mathcal{B}(p_i)$. Finally, the sensor output was generated, as a function of its state s_i, the true output y_i, and the accuracy σ_i. If $s_i = 1$ (the sensor is in good operating condition), z_i was sampled from a Gaussian distribution with mean y_i and standard deviation σ_i:

$$z_i \sim \mathcal{N}(y_i, \sigma_i)$$

If however the sensor was broken ($s_i = 0$), its output z_i was sampled from a uniform distribution on the whole output domain:

$$z_i \sim \mathcal{U}_{[0,10]}.$$

The overall distribution of z_i is thus a mixture of a Gaussian and a uniform distribution, with proportions p_i and $1-p_i$, respectively:

$$z_i \sim p_i \mathcal{N}(y_i, \sigma_i) + (1-p_i)\mathcal{U}_{[0,10]} \tag{41}$$

In summary, the data generation procedure thus consists of the following steps, repeated for each example $i \in \{1, \ldots, N\}$:

1. compute the input x_i using (37);
2. compute the true output y_i using (38);
3. generate the noise standard deviation σ_i and the reliability parameter p_i using (39) and (40), respectively;
4. generate the measurement value z_i using (41).

One hundred data sets $\mathcal{L}^{(\ell)}$, $\ell = 1, \ldots, 100$ were generated using this procedure, four of which are shown in Figure 4.

Methods and Results

The EVREG method was applied to this data, and compared with the NW and k-NN methods.

In EVREG, each training example $(x_i, z_i, \sigma_i, p_i)$ was encoded as a pair (x_i, m_i), with the FBA m_i defined as:

$$\begin{aligned} m_i(F_i) &= p_i \\ m_i(\mathcal{Y}) &= 1 - p_i, \end{aligned}$$

where F_i is a Gaussian fuzzy number with center z_i and standard deviation σ_i:

$$F_i(u) = \exp\left(-\frac{1}{2}\frac{(u-z_i)^2}{\sigma_i^2}\right), \quad u \in \mathbb{R}.$$

Hence, the FBA m_i encodes the measurement value z_i, together with its imprecision σ_i and reliability p_i.

The discounting function ϕ in (15) was defined as:

$$\phi(|x - x_i|) = 0.99 \exp\left(-\frac{(x-x_i)^2}{\theta^2}\right)$$

and parameter θ was optimized using the procedure described in Section 4.2. We used the k nearest neighbor version of the method defined by (20), with $k = 5$.

For each training set $\mathcal{L}_\ell$, the error was computed as the mean squared differences between the true output y_i and $\widehat{y}_i^{(\ell)}$, the pignistic expectation (22) of $m_{y_i}[x_i, \mathcal{L}^{(\ell)}]$:

$$\mathrm{err}^{(\ell)} = \frac{1}{N} \sum_{i=1}^{N} (y_i - \widehat{y}_i^{(\ell)})^2. \tag{42}$$

The average error over the L=100 trial was then defined as :

$$\overline{\mathrm{err}} = \frac{1}{L} \sum_{\ell=1}^{L} \mathrm{err}^{(\ell)} \tag{43}$$

The performances of our method were compared to those of the NW and k-NN methods. The tuning parameters of these methods (Gaussian kernel bandwidth for the NW method, and k for the k-NN method) were optimized by leave-one-out cross-validation for each training set. When faced to uncertain data using such classical techniques, the two options are either to keep all the data, or to discard the most unreliable data. These two strategies were simulated by applying each of the two classical methods in three different conditions:

1. using all the training data;
2. using only the training examples i such that $p_i > 0.2$;
3. using only the training examples i such that $p_i > 0.5$.

Figure 5 displays the results obtained for one particular data set. As shown by this example, the EVREG model is able to take advantage of all the information in the training set, including the imprecision and reliability values (see Fig. 5(a)). When applied to the same data, the k-NN and NW smoothers perform poorly, which can be explained by the fact that they are not able to use the imprecision and reliability information (Fig. 5(b)). One way to introduce part of this information is to discard the most unreliable data points, which however only marginally improves the results in this case (Fig. 5(c) and (d)).

These observations are confirmed by looking at the overall results for the 100 trials. The average errors (43) for each of the six methods tested are displayed in Figure 6.

The EVREG method has the least average error, whereas the other six methods are roughly equivalent. Figure 7 shows boxplots of the differences between the error of each classical method and the error of the EVREG model, for the 100 trials. As shown by this graphical representation, the superiority of the FBA-based method is highly significant.

5.3 Two Unreliable Sensors

Problem Description and Data Generation

This example continues the previous one, assuming that we now have two sensors S^1 and S^2 of different time-varying accuracy and reliability. As before, the input values x_i are fixed and defined by (37), and the true outputs are computed as a function of the inputs according to (38). Each sensor S^j $(j = 1, 2)$ provides for each input value x_i a random measurement z_i^j from the following distribution:

$$z_i^j \sim p_i^j \mathcal{N}(y_i, \sigma_i^j) + (1 - p_i^j)\mathcal{U}_{[0,10]},$$

where p_i^j is the probability that sensor S^j is in good operating condition, and σ_i^j is the standard deviation of the measurement noise for the same sensor. Parameters p_i^j and σ_i^j $(i = 1, \ldots, N,\ j = 1, 2)$ are generated randomly using the same uniform distributions as in the previous section (39-40).

Each learning example is thus of the form

$$(x_i, z_i^1, \sigma_i^1, p_i^1, z_i^2, \sigma_i^2, p_i^2).$$

Using the classical regression approach, a learning example can only be of the form $(x_i, z_i) \in \mathbb{R}^2$. In that case, it is quite natural to define z_i as a weighted sum of z_i^1 and z_i^2, the weights being equal to the reliability parameters:

$$z_i = \frac{p_i^1 z_i^1 + p_i^2 z_i^2}{p_i^1 + p_i^2}.$$

In the EVREG model, a simple way to encode the leaning information is to represent the output of each sensor S^j for input x_i by a FBA m_i^j defined as:

$$\begin{aligned} m_i^j(F_i^j) &= p_i^j \\ m_i^j(\mathcal{Y}) &= 1 - p_i^j, \end{aligned}$$

where F_i^j is a Gaussian fuzzy number with center z_i^j and standard deviation σ_i^j:

$$F_i^j(u) = \exp\left(-\frac{1}{2}\frac{(u - z_i^j)^2}{\sigma_i^j}\right), \quad u \in \mathbb{R}.$$

The outputs of the two sensors are then combined using the Dempster's rule (conjunctive sum followed by smooth normalization) to form a new FBA $m_i = m_i^1 \oplus m_i^2$.

Results

For this learning task, we have compared the performances of EVREG to those of the NW method (the NW and k-NN predictors were shown in Section 5.2 to yield quite similar results). As before, parameter λ in the NW method, and parameter θ in EVREG were optimized using leave-one-out cross-validation. The experiment (including random data generation) was repeated 100 times. For both methods, the error $\mathrm{err}^{(\ell)}$ for training set $\mathcal{L}^{(\ell)}$ was computed using (42), and the average error over the 100 learning sets was computed using (43). The average error was 5.96 for our method vs. 8.00 for the NW predictor. The 99% confidence interval for the difference between the mean errors of the NW and EVREG models is $[1.21, 2.86]$, meaning that the observed difference is significant at the 0.01 level.

A boxplot of the distribution of error differences between the two methods for the 100 trials is shown in Figure 8, and results for a particular learning set are shown in Figure 9.

6 Conclusion

A new nonparametric regression technique has been described. This technique, called EVREG, is rooted in Belief Function Theory, and makes extensive use of two fundamental operations in this theory: discounting and Dempster's rule of combination. The EVREG model differs from both standard statistical nonparametric regression models and fuzzy systems in two important respects:

- the response variable y_i for learning examples is allowed to be partially known, and specified in the form of a crisp or fuzzy belief assignment m_i; this type of data thus generalizes both numerical, interval-valued and fuzzy data types usually considered in conventional statistics or fuzzy data analysis;

- the model output for an input $\mathbf{x}$ is also given in the form of crisp or fuzzy belief assignment, which quantifies the uncertainty on the response variable, resulting from both the imperfection of learning data, and the dissimilarity of $\mathbf{x}$ to known examples in the learning set.

Simulations have demonstrated the good performances of EVREG in realistic situations in which the observations are acquired from one or several sensors with limited accuracy and reliability.

References

[1] A. P. Dempster. Upper and lower probabilities induced by a multivalued mapping. *Annals of Mathematical Statistics*, AMS-38:325–339, 1967.

[2] T. Denœux. A k-nearest neighbor classification rule based on Dempster-Shafer theory. *IEEE Trans. on Systems, Man and Cybernetics*, 25(05):804–813, 1995.

[3] T. Denœux. Application du modèle des croyances transférables en reconnaissance de formes. *Traitement du Signal*, 14(5):443–451, 1998.

[4] T. Denœux. Modeling vague beliefs using fuzzy-valued belief structures. *Fuzzy Sets and Systems*, 116(2):167–199, 2000.

[5] T. Denœux. A neural network classifier based on Dempster-Shafer theory. *IEEE Trans. on Systems, Man and Cybernetics A*, 30(2):131–150, 2000.

[6] T. Denœux. Inner and outer approximation of belief structures using a hierarchical clustering approach. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, 9(4):437–460, 2001.

[7] T. Denœux and L. M. Zouhal. Handling possibilistic labels in pattern classification using evidential reasoning. *Fuzzy Sets and Systems*, 122(3):47–62, 2001.

[8] Ph. Diamond and H. Tanaka. Fuzzy regression analysis. In R. Slowinski, editor, *Fuzzy sets in decision analysis, operations research and statistics*, pages 349–387. Kluwer academic Publishers, Norwell, 1998.

[9] D. Dubois and Prade H. *Possibility theory: an approach to computerized processing of uncertainty.* Plenum Press, New York, 1988.

[10] D. Dubois and H. Prade. Decision evaluation methods under uncertainty and imprecision. In V. Kacprzyk and M. Fedrizzi, editors, *Combining fuzzy imprecision with probabilistic uncertainty in decision making*, pages 48–65. Springer Verlag, Berlin, 1989.

[11] S. Fabre, A. Appriou, and X. Briottet. Presentation and description of two classification methods using data fusion based on sensor management. *Information Fusion*, 2:49–71, 2001.

[12] W. Härdle. *Applied nonparametric regression.* Cambridge University Press, Cambridge, 1990.

[13] T. Hastie, R. Tibshirani, and J. Friedman. *The elements of statistical learning.* Springer, New-York, 2001.

[14] T. J. Hastie and R. J. Tibshirani. *Generalized additive models.* Chapman and Hall, New York, 1990.

[15] J.S.R Jang, C.T. Sun, and E. Mizutani. *Neuro-Fuzzy and Soft Computing.* Prentice-Hall, Upper Saddle River, N.J, 1997.

[16] S. Petit-Renaud. *Application de la théorie des croyances et des systèmes flous à l'estimation fonctionnelle en présence d'informations incertaines ou imprécises (in French).* PhD thesis, Université de Technologie de Compiègne, 1999.

[17] S. Petit-Renaud and T. Denœux. Handling different forms of uncertainty in regression analysis: a fuzzy belief structure approach. In A. Hunter and S. Pearsons, editors, *Symbolic and quantitative approaches to reasoning and uncertainty (ECSQARU'99)*, pages 340–351, London, June 1999. Springer Verlag.

[18] S. Petit-Renaud and T. Denœux. Regression analysis using fuzzy evidence theory. In *Proceedings of FUZZ-IEEE'99*, volume 3, pages 1229–1234, Seoul, August 1999.

[19] G. Shafer. *A mathematical theory of evidence*. Princeton University Press, Princeton, N.J., 1976.

[20] P. Smets. The Transferable Belief Model for quantified belief representation. In D. M. Gabbay and P. Smets, editors, *Handbook of Defeasible reasoning and uncertainty management systems*, volume 1, pages 267–301. Kluwer Academic Publishers, Dordrecht, 1998.

[21] Ph. Smets. The degree of belief in a fuzzy event. *Information Sciences*, 25:1–19, 1981.

[22] Ph. Smets. Varieties of ignorance. *Information Sciences*, 57-58:135–144, 1991.

[23] Ph. Smets. What is Dempster-Shafer's model? In *Advances in the Dempster-Shafer theory of evidence*, pages 5–34. Wiley, 1994.

[24] Ph. Smets and R. Kennes. The Transferable Belief Model. *Artificial Intelligence*, 66:191–243, 1994.

[25] T Takagi and M. Sugeno. Fuzzy identification of systems and its applications to modeling and control. *IEEE Transactions on Systems, Man and Cybernetics*, 15:116–132, 1985.

[26] R. R. Yager. Generalized probabilities of fuzzy events from fuzzy belief structures. *Information Sciences*, 28:45–62, 1982.

[27] R. R. Yager. On the normalization of fuzzy belief structure. *International Journal of Approximate Reasoning*, 14:127–153, 1996.

[28] R. R. Yager and D. P. Filev. Including probabilistic uncertainty in fuzzy logic controller modeling using Dempster-Shafer theory. *IEEE Transactions on Systems, Man and Cybernetics*, 25(8):1221–1230, 1995.

[29] J. Yen. Generalizing the Dempster-Shafer theory to fuzzy sets. *IEEE Transactions on Systems, Man and Cybernetics*, 20(3):559–569, 1990.

[30] L. A. Zadeh. Fuzzy sets and information granularity. In R. K. Ragade M. M. Gupta and R. R. Yager, editors, *Advances in Fuzzy Sets Theory and Applications*, pages 3–18. North-Holland Publishing Co., 1979.

[31] L. M. Zouhal and T. Denœux. An evidence-theoretic k-NN rule with parameter optimization. *IEEE Trans. on Systems, Man and Cybernetics C*, 28(2):263–271, 1998.

[32] R. Zwick, E. Carlstein, and D. V. Budescu. Measures of similarity among fuzzy concepts: a comparative analysis. *International Journal of Approximate Reasoning*, 1:221–242, 1987.

Figures

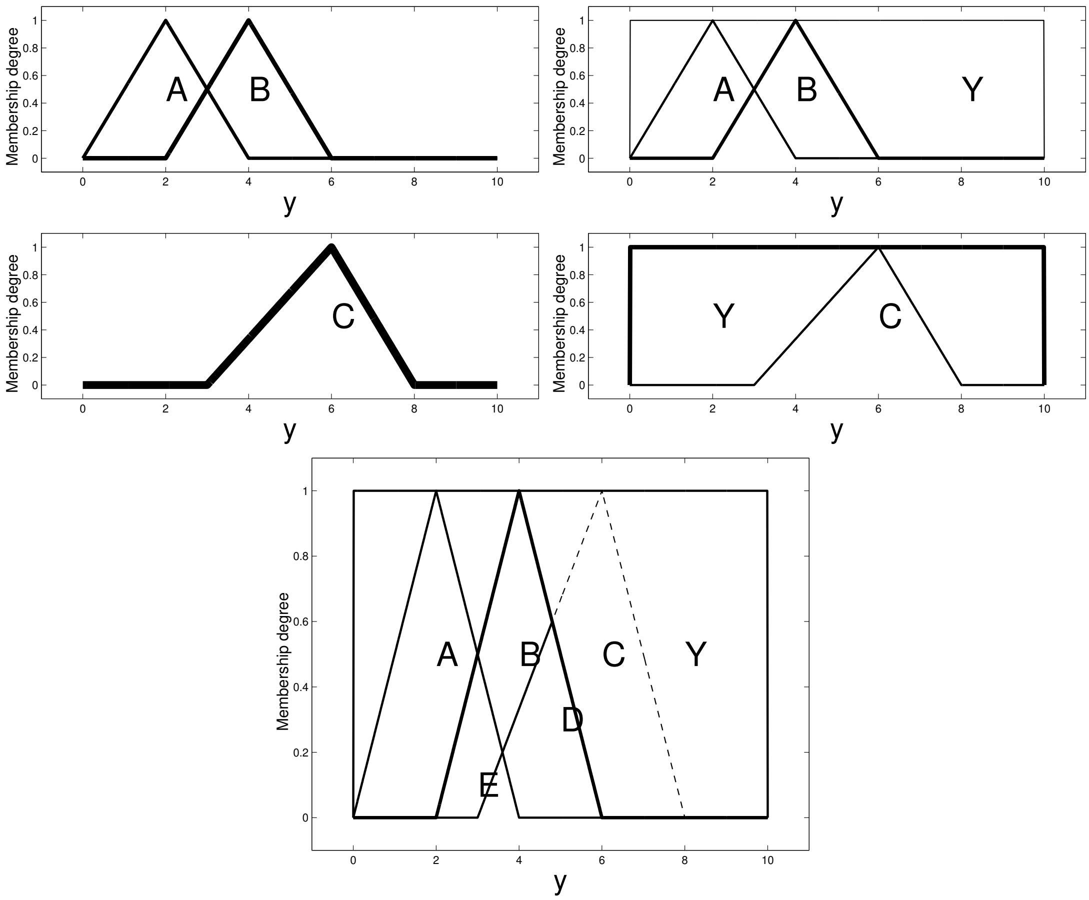

Figure 1: Example 1: calculation of $m_y[\mathbf{x}, \mathcal{L}]$ from two elements of a training set $\{(\mathbf{x}_1, m_1), (\mathbf{x}_2, m_2)\}$). Upper left: m_1, defined by two fuzzy focal elements A and B, and m_2, with a single fuzzy focal element C. Upper right: $m_y[\mathbf{x}, e_1]$ and $m_y[\mathbf{x}, e_2]$. Bottom: $m_y[\mathbf{x}, e_1, e_2]$ with $\mathcal{F}(m_y[\mathbf{x}, e_1, e_2]) = \{A, B, C, \mathcal{Y}, D, E\}$, where $E = A \cap C$ and $D = B \cap C$. The thickness of the lines is proportional to the masses.

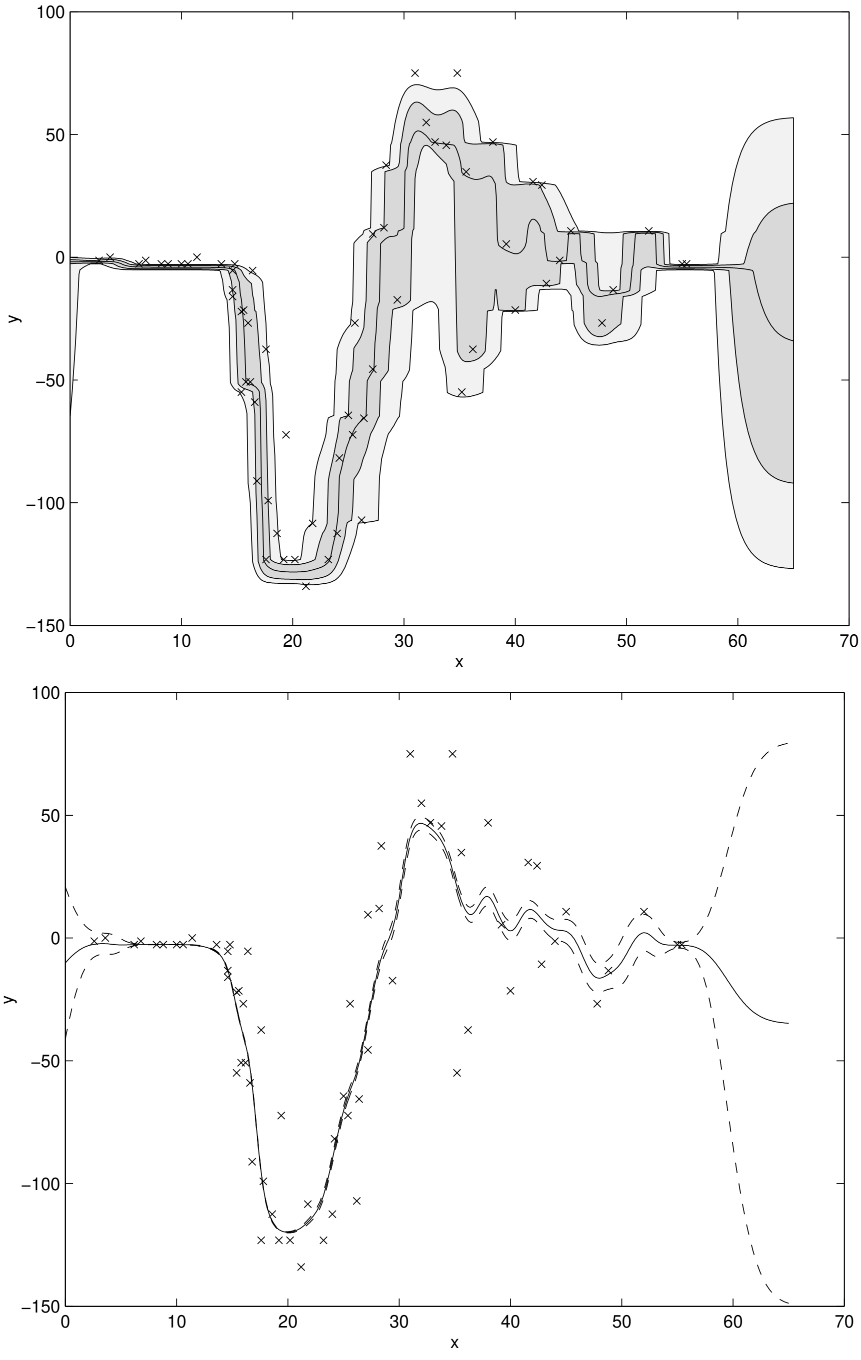

Figure 2: Motorcycle data. Up: training data (x) with 0.1, 0.25, 0.5, 0.75 and 0.90 quantiles. Down: training data (x) with pignistic expectation (solid line) and upper and lower expectations (broken lines).

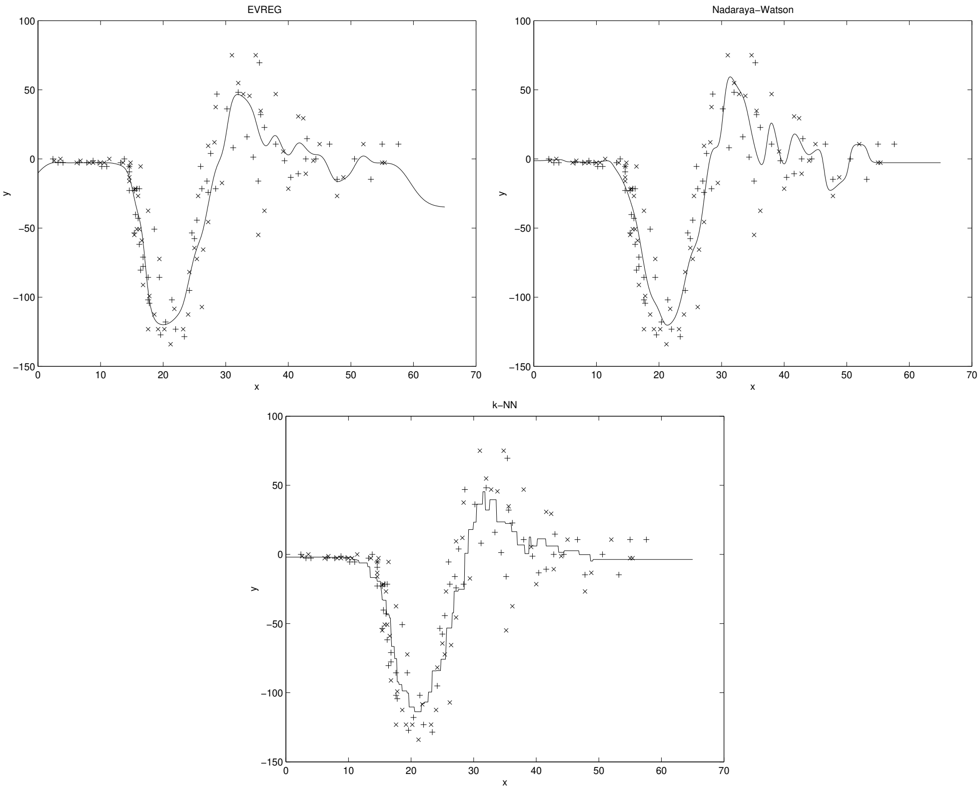

Figure 3: Motorcycle data: prediction curves for the EVREG (upper left), Nadaraya-Watson (upper right) and k-NN (down) methods, together with the training (x) and test (+) data.

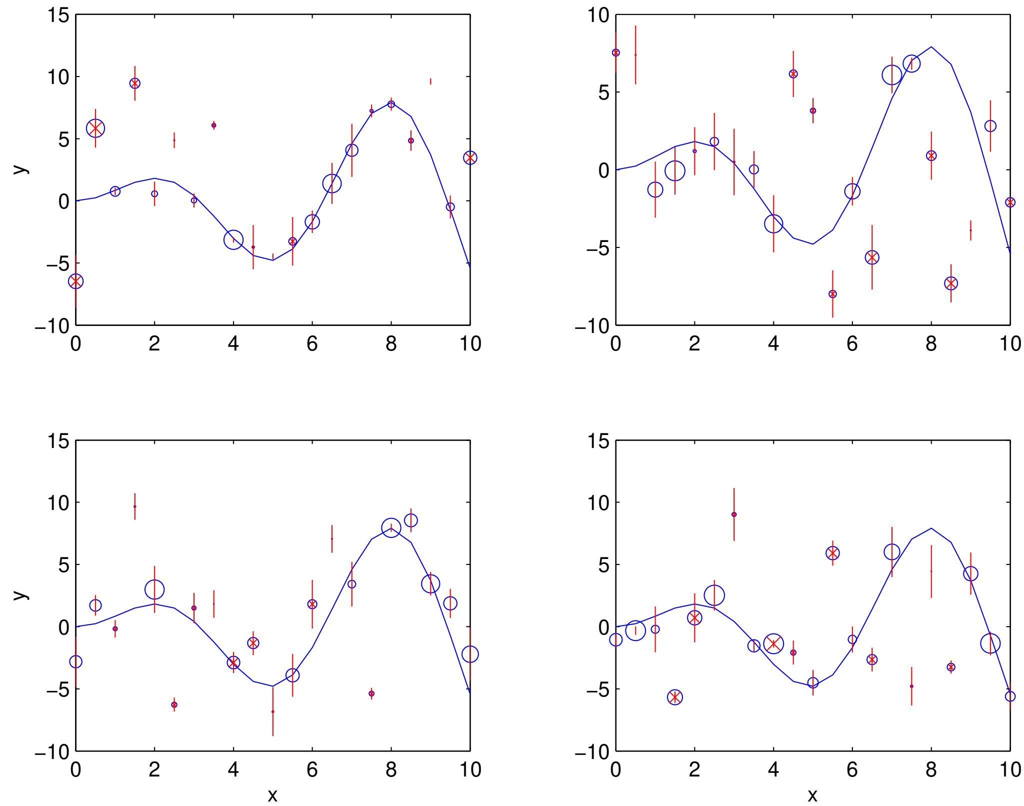

Figure 4: Four generated data sets for the unreliable sensor experiment. The solid curve indicates the true output values y_i as a function of the inputs x_i. The circles show the simulated measurements z_i. The radius of each circle is proportional to the reliability index p_i. A cross inside the circle indicates that the sensor was broken ($s_i = 0$). Each vertical segment represents the interval $[z_i - \sigma_i, z_i + \sigma_i]$.

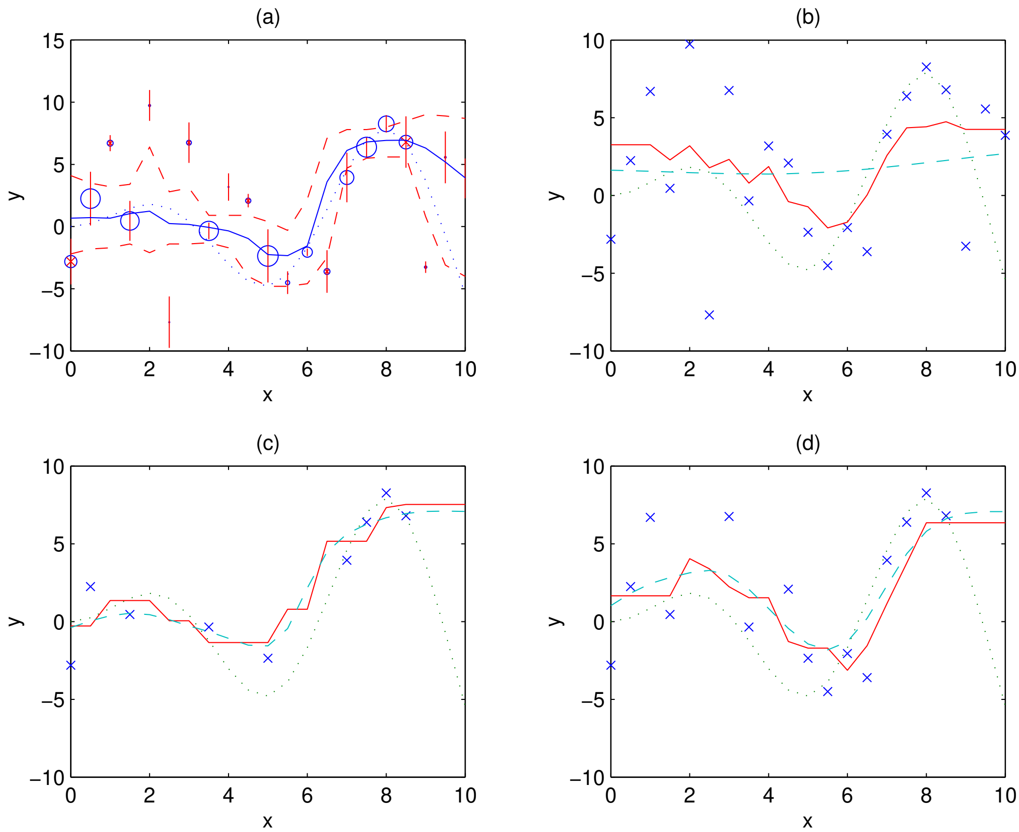

Figure 5: Unreliable sensor experiment: (a) point prediction (solid line) and first and ninth deciles of the pignistic distribution (broken lines) obtained by EVREG, with the data set (plotted as in Figure 4) and the true outputs (dotted line); (b) predictions obtained using the k-NN (solid line) and NW (broken line) regressors; (c) predictions obtained using the k-NN (solid line) and NW (broken line) regressors, using only examples i such that $p_i > 0.5$; (d) predictions obtained using the k-NN (solid line) and NW (broken line) regressors, using only examples i such that $p_i > 0.2$.

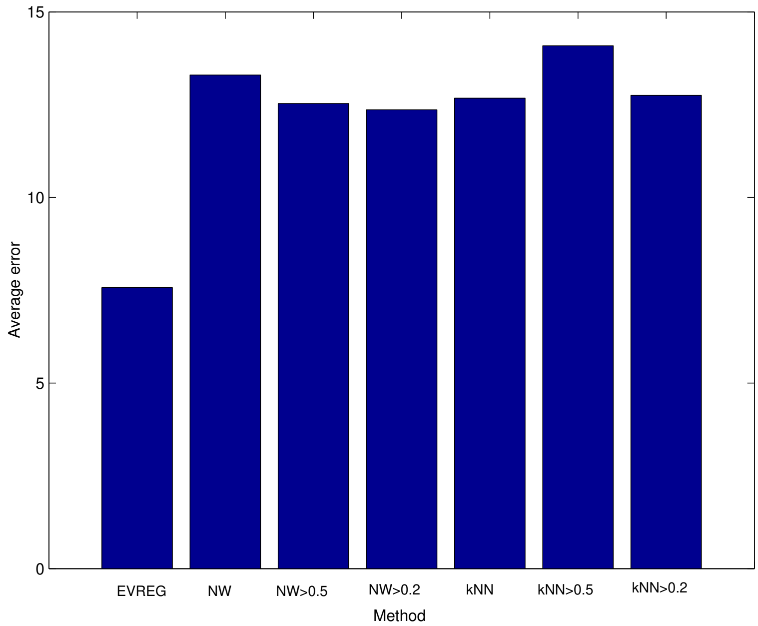

Figure 6: Unreliable sensor experiment: average errors (over the 100 trial) for EVREG, and six classical approaches: the Nadaraya-Watson (NW) and k-NN regressors, in 3 learning conditions (with all training data, with examples i such that $p_i > 0.5$, with examples i such that $p_i > 0.2$).

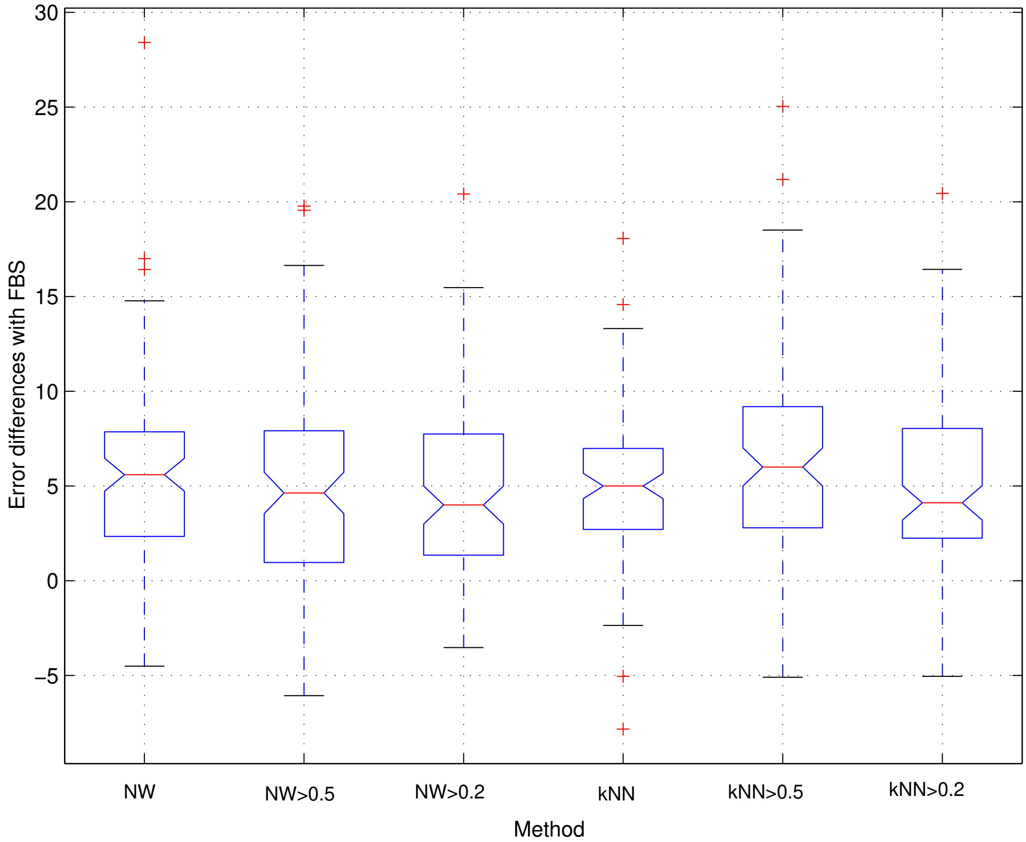

Figure 7: Unreliable sensor experiment: boxplots of the distributions of error differences between each of the 6 classical methods, and the EVREG model. The horizontal line inside each box indicates the median, while the lower and upper lines of the box are the 25th and 75th percentiles of the sample.

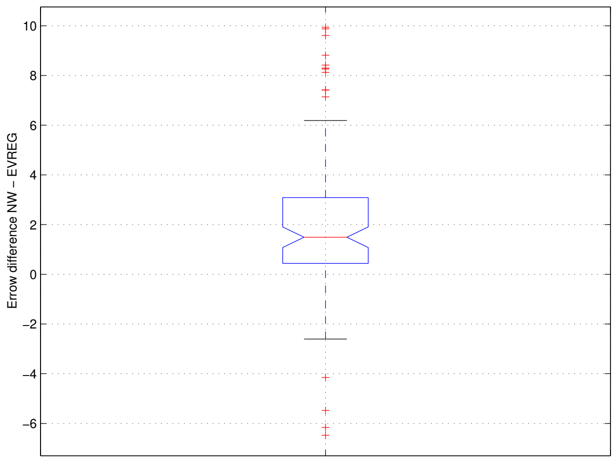

Figure 8: Two unreliable sensors experiment: boxplot of the distributions of error differences between the NW and EVREG models.

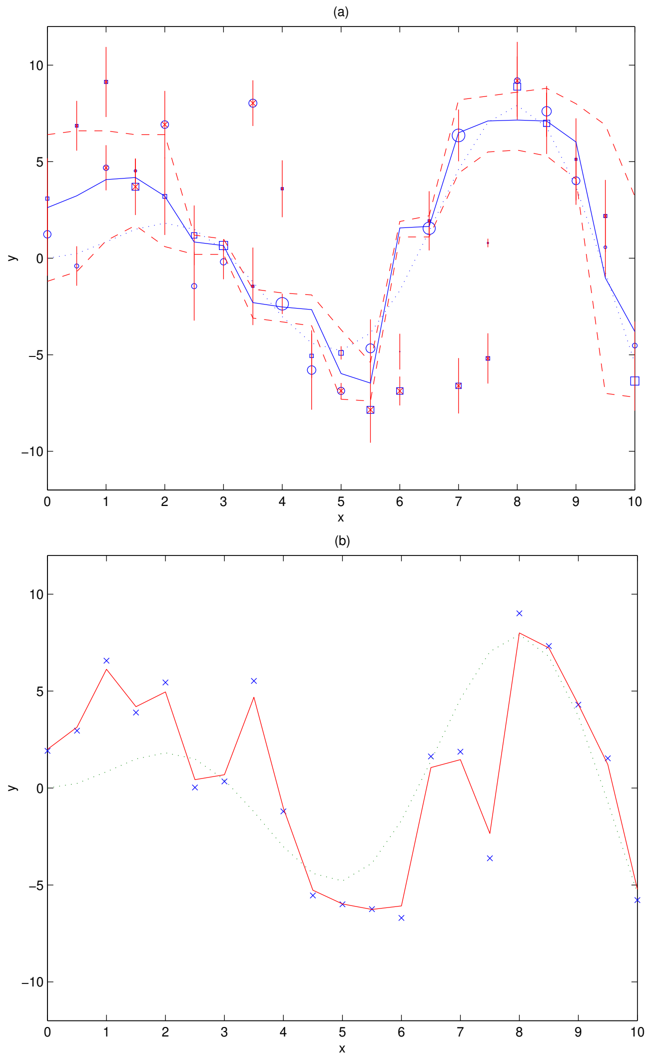

Figure 9: Two unreliable sensors experiment: (a) true output (dotted line), point prediction (solid line) and first and ninth deciles of the pignistic distribution (broken lines) obtained by EVREG, with the data set (the outputs from sensors S^1 and S^2 are shown as circles and squares, respectively); (b) true output (dotted line) and predictions obtained using the NW method (solid line).