View metadata, citation and similar papers at core.ac.uk

brought to you by CORE

provided by Aberystwyth Research Portal

Aberystwyth University

Fuzzy Rough Positive Region based Nearest Neighbour Classification

Verbiest, Nele; Cornelis, Chris; Jensen, Richard

Published in:
2012 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)

Publication date:
2012

Citation for published version (APA):
Verbiest, N., Cornelis, C., & Jensen, R. (2012). Fuzzy Rough Positive Region based Nearest Neighbour Classification. In *2012 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)* (pp. 1961-1967). IEEE Press.

General rights
Copyright and moral rights for the publications made accessible in the Aberystwyth Research Portal (the Institutional Repository) are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

• Users may download and print one copy of any publication from the Aberystwyth Research Portal for the purpose of private study or research.
• You may not further distribute the material or use it for any profit-making activity or commercial gain
• You may freely distribute the URL identifying the publication in the Aberystwyth Research Portal

Take down policy
If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.

tel: +44 1970 62 2400
email: is@aber.ac.uk

Download date: 09. Jul. 2020

Fuzzy Rough Positive Region based Nearest Neighbour Classification

Nele Verbiest*, Chris Cornelis*†, Richard Jensen‡
*Department of Applied Mathematics and Computer Science
Ghent University, Belgium
Email: Nele.Verbiest@UGent.be
Email: chriscornelis@ugr.es
†Department of Computer Science and Artificial Intelligence
University of Granada, Spain
‡Department of Computer Science
Aberystwyth University, Ceredigion, Wales, UK
Email: rkj@aber.ac.uk

***Abstract*—This paper proposes a classifier that uses fuzzy rough set theory to improve the Fuzzy Nearest Neighbour (FNN) classifier. We show that previous attempts to use fuzzy rough set theory to improve the FNN algorithm have some shortcomings and we overcome them by using the fuzzy positive region to measure the quality of the nearest neighbours in the FNN classifier. A preliminary experimental evaluation shows that the new approach generally improves upon existing methods.**

I. Introduction

Classification, the problem of labeling an instance based on previously seen data, is one of the main subjects in machine learning and pattern recognition, with applications in fields like spam detection, natural language processing, speech recognition, bioinformatics and many more.

One of the best-known classifiers is the K Nearest Neighbour (KNN) classifier [1]. It assigns a target instance to the class most common among its nearest neighbours. Due to its low bias, understandability and the fact that it does not impose any assumptions on the data, KNN is widely used in many applications.

However, KNN faces the problem that each of its neighbours is considered equally important, independent of the neighbour's distance to the target instance. To overcome this problem, Keller et al. suggested in [2] to introduce fuzzy set theory [3] into the classical KNN decision rule. By means of the fuzzy similarity relation, instances can now partially belong to the set of nearest neighbours and are weighted accordingly.

This Fuzzy Nearest Neighbour (FNN) method improves the KNN method substantially, but in [4] it was noted that this classifier cannot adequately handle imperfect knowledge. In particular, when every training pattern is far removed from the test object, and hence there are no suitable neighbours, the algorithm is still forced to make clear-cut predictions. This is because the predicted membership degrees to the various decision classes always need to sum up to 1. To address this problem, Sarkar [4] introduced a so-called fuzzy rough ownership function that, when plugged into the conventional FNN algorithm, produces class confidence values that do not necessarily sum up to 1. However, this method does not refer to the main ingredients of rough set theory, i.e., lower and upper approximation.

Therefore, in [5], [6] the authors proposed a nearest neighbour algorithm that measures the extent to which the nearest neighbours belong to the fuzzy lower and upper approximations of a certain class to predict the class of the target instance. In order to deal with noisy data, they took this fuzzy rough nearest neighbour (FRNN) approach one step further and used vaguely quantified rough sets (VQRSs, [7]).

In this paper, we show that FRNN only takes into account one instance, and that the VQNN classifier is similar to the FNN classifier if they use the same similarity relation. As a result, none of the existing fuzzy nearest neighbour methods are able to use fuzzy rough set theory to improve the FNN classifier.

Fuzzy rough set theory was designed to model imperfect knowledge, that is, to take into account the fact that there are instances similar to each other but belonging to different classes. This occurs for border instances, instances in overlapping classes or mislabeled instances. The fuzzy positive region measures for each instance the extent to which instances similar to it belong to the same class. As a result, the fuzzy positive region can be used to measure the quality of an instance.

In this work, we improve the FNN classifier by weighting the nearest neighbour instances according to their membership degree to the fuzzy positive region. In other words, when classifying a target instance, we do not only take into account the distance from the target instance to its nearest neighbours, but also the quality of the nearest neighbours.

The remainder of this paper is structured as follows: in Section II, we review the KNN, FNN, FRNN and VQNN techniques. In Section III, we introduce our new classifier, called POSNN, and in Section IV, we experimentally evaluate POSNN. We conclude and highlight future research directions in Section V.

II. Related Work

In this section we briefly recall the KNN technique and some of its fuzzy (rough) extensions. We assume that we are given a training dataset X and a target instance t for which we want to predict the class. The training data consists of n instances $x_1, \ldots, x_n$ and covers m features $a_1, \ldots, a_m$ that can be nominal or continuous. The value of an attribute a for an instance x is denoted by $a(x)$, the class value of x is denoted by $d(x)$.

A. K Nearest Neighbour (KNN)

The KNN classifier [1] determines the K instances in X closest to t and then assigns t to the class that is best represented among these K neighbours. In case of ties, a class is assigned at random from the candidate classes. Usually, the following Euclidean distance is used:

$$d_{eucl}(x,t) = \sqrt{\sum_{i=1}^{m} d^i_{eucl}(x,t)},$$

where

$$d^i_{eucl}(x,t) = (a_i(x) - a_i(t))^2$$

if a_i is a continuous feature and

$$d^i_{eucl}(x,t) = \begin{cases} 1 & \text{if } a_i(x) \neq a_i(t) \\ 0 & \text{if } a_i(x) = a_i(t) \end{cases}$$

if a_i is nominal.
KNN is a simple classification method that does not impose assumptions on the data. Due to its local nature it has low bias, more specifically, the error rate of 1NN asymptotically never exceeds twice the optimal Bayes error rate [8].

B. Fuzzy Nearest Neighbour (FNN)

One drawback of the KNN classifier is that each of the selected neighbours is considered equally important when assigning a class to the target instance. This problem is illustrated in Figure 1.

Consider KNN with K=10. The target instance is represented by a dot. There are 2 classes among the 10 neighbours, 4 neighbours of the first class (represented by a cross) are lying close to the target instance, while 6 instances from the second class (represented by a diamond) are lying further from the target instance. Due to the simple voting strategy of KNN, the target instance will be assigned to the second class, although it is clear from the figure that the first class should be preferred. Moreover, the classification highly depends on the number of neighbours chosen.

In [2], Keller et al. address this problem by incorporating fuzzy set theory. More specifically, they first determine the K nearest neighbours NN of the target instance t and then assign t to the class C for which the following sum is maximal:

$$\frac{\sum_{x \in NN} R(x,t)C(x)}{\sum_{x \in NN} R(x,t)}.$$

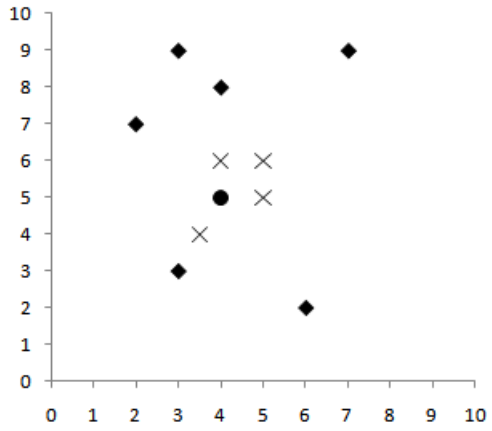

Fig. 1. KNN with K=10 assigns the target instance (represented by a dot) to the majority class (represented by diamonds), but it is clear that the target instance should be assigned to the minority class (represented by crosses).

The similarity $R(x,t)$ between x and t is defined as follows:

$$R(x,t) = \frac{1}{d_{eucl}(x,t)^{\frac{2}{m-1}}}. \quad (1)$$

The parameter m determines the extent to which the distance is weighted when calculating each neighbour's contribution. We use $m = 2$ throughout the paper, as suggested in [2].

We consider two options to define the class membership function C. The basic approach is that $C(x)$ is 1 if the class of x is C and 0 otherwise. The second approach, proposed by Keller et al. in [2], defines $C(x)$ as

$$0.51 + \frac{n_C}{K} * 0.49$$

if x is in class C and

$$\frac{n_C}{K} * 0.49$$

otherwise. Here, n_C is the number of instances among the K nearest neighbours of x which belong to class C. Using this definition, the class membership $C(x)$ will be more than 0.51 if x belongs to class C, and the class membership is higher if many of the neighbours belong to class C. On the other hand, $C(x)$ is smaller than 0.49 if x does not belong to class C, and is smaller if fewer instances belong to class C.

We refer to the first class membership as C_{crisp} and to the second as $C_{gradual}$.

For the example in Figure 1, FNN will assign a higher weight in the voting strategy to the four instances of the minority class and the target instance will be assigned to the minority class.

In the following subsections, we present previous attempts to improve FNN by means of fuzzy rough sets.

C. Fuzzy Rough Nearest Neighbour (FRNN)

In [6], the authors use the fuzzy lower and upper approximation to improve the FNN algorithm. The FRNN algorithm assigns a class to a target instance t as follows:

- Determine NN, the set of K nearest neighbours of t
- Assign t to the class C for which

$$(R \downarrow C)(t) + (R \uparrow C)(t)$$

is maximal.

Here, the lower and upper approximation [9] are defined over the nearest neighbours of t:

$$(R \downarrow C)(t) = \min_{x \in NN} \mathcal{I}(R(x,t), C(x))$$
$$(R \uparrow C)(t) = \max_{x \in NN} \mathcal{T}(R(x,t), C(x))$$

The class membership is 1 if x is in class C and 0 otherwise[1]. We define the similarity relation R as suggested by the authors in [6]:

$$R(x,t) = \min_{i \in \{1,\ldots,n\}} R_{a_i}(x,t), \tag{2}$$

with

$$R_{a_i}(x,t) = 1 - \frac{|a_i(x) - a_i(t)|}{|a_i^{max} - a_i^{min}|},$$

where a_i^{max} and a_i^{min} are the maximal and minimal occurring value of the attribute a_i.

The fuzzy implicator $\mathcal{I}$ is a mapping $[0,1]^2 \to [0,1]$ that is decreasing in the first and increasing in the second argument, and for which $\mathcal{I}(0,0) = 1$ and $(\forall x \in [0,1])(\mathcal{I}(1,x) = x)$. The fuzzy t-norm $\mathcal{T}$ is a commutative and associative mapping $[0,1]^2 \to [0,1]$ that is increasing in both arguments, and for which $(\forall x \in [0,1])(\mathcal{T}(1,x) = x)$. In this paper, we use the Kleene-Dienes implicator $\mathcal{I}_{KD}$ and minimum t-norm $\mathcal{T}_M$, defined as follows for $x, y \in [0,1]$:

$$\mathcal{I}_{KD}(x,y) = \max(1-x, y)$$
$$\mathcal{T}_M(x,y) = \min(x,y).$$

The fuzzy rough lower approximation $(R \downarrow C)$ of t expresses to which extent instances similar to t are in class C. The fuzzy rough upper approximation $(R \uparrow C)$ of t expresses to which extent there exist instances that are similar to t and belong to class C. As a result, $(R \downarrow C)(t) + (R \uparrow C)(t)$ provides a good clue to predict if t belongs to C.

When we look in more detail at this method, we see that only one instance influences the outcome, as proven in the next theorem:

Theorem 1: FRNN assigns a target instance t to the class of the instance $x \in NN$ for which $R(x,t)$ is maximal.

Proof: We first rewrite the lower approximation of the target instance (we use the fact that $\mathcal{I}(a,1) = 1$ for all $a \in [0,1]$):

$$\begin{aligned}
&(R \downarrow C)(t) \\
&= \min_{x \in NN} \mathcal{I}(R(x,t), C(x)) \\
&= \min_{x \in NN, C(x)=0} \mathcal{I}(R(x,t), C(x)) \\
&= \min_{x \in NN, C(x)=0} (1 - R(x,t)) \\
&= 1 - \max_{x \in NN, C(x)=0} R(x,t)
\end{aligned}$$

[1]We could use the $C_{gradual}$ definition, but we use the definition as in [6]

Here, we assumed that there was at least one instance x for which $C(x) = 0$. In case there is no such instance, all instances belong to the same class and then the theorem obviously holds.

Next, we consider the upper approximation of the target instance:

$$\begin{aligned}
&(R \uparrow C)(t) \\
&= \max_{x \in NN} \mathcal{T}(R(x,t), C(x)) \\
&= \max_{x \in NN, C(x)=1} \mathcal{T}(R(x,t), C(x)) \\
&= \max_{x \in NN, C(x)=1} R(x,t)
\end{aligned}$$

This result only holds if there is at least one instance for which $C(x) = 1$. Otherwise, $(R \downarrow C)(t) = 0$.

In case there is at least one instance x for which $C(x) = 1$, t is assigned to the class for which the following expression is maximal:

$$\max_{x \in NN, C(x)=1} R(x,t) - \max_{x \in NN, C(x)=0} R(x,t)$$

Now suppose that $y \in NN$ is the instance for which

$$\max_{x \in NN} R(y,t)$$

is maximal, and suppose D is the class of y, i.e., $D(y) = 1$. Then for each other class C,

$$\max_{x \in NN, C(x)=1} R(x,t)$$

is strictly smaller than

$$\max_{x \in NN, D(x)=1} R(x,t)$$

and

$$\max_{x \in NN, C(x)=0} R(x,t)$$

is strictly larger than

$$\max_{x \in NN, D(x)=0} R(x,t).$$

In case there is no instance for which $C(x) = 1$, t is assigned to the class for which the following expression is maximal:

$$-\max_{x \in NN, C(x)=0} R(x,t)$$

Again, suppose that $y \in NN$ is the instance for which

$$\max_{x \in NN} R(y,t)$$

is maximal, and suppose D is the class of y, i.e., $D(y) = 1$. Then for each other class C,

$$\max_{x \in NN, C(x)=0} R(x,t)$$

is strictly larger than

$$\max_{x \in NN, D(x)=0} R(x,t).$$

As a result, the target instance t is assigned to class D, which is the nearest neighbour of t with respect to the similarity measure R. ■

Theorem 1 shows that the classification of t is only determined by one instance, and that the concepts of fuzzy rough set theory are not fully exploited.

D. Vaguely Quantified Nearest Neighbour (VQNN)

Another approach suggested in [6] to incorporate fuzzy rough set theory is using VQRSs instead of the basic fuzzy rough sets.

The resulting VQNN algorithm assigns a class to a target instance t as follows:

- Determine NN, the K nearest neighbours of t
- Assign t to the class C for which

$$(R \downarrow^{Q_u} C)(t) + (R \uparrow^{Q_l} C)(t)$$

is maximal.

Here, the upper and lower approximations of VQ rough sets are defined as follows:

$$(R \downarrow^{Q_u} C)(t) = Q_u(\frac{\sum\limits_{x \in NN} \min(R(x,t), C(x))}{\sum\limits_{x \in NN} R(x,t)})$$

$$(R \downarrow^{Q_l} C)(t) = Q_l(\frac{\sum\limits_{x \in NN} \min(R(x,t), C(x))}{\sum\limits_{x \in NN} R(x,t)})$$

The operators Q_u and Q_l are fuzzy quantifiers that represent *most* and *some* respectively, i.e., they are increasing $[0,1] \rightarrow [0,1]$ mappings such that $Q_u(1) = Q_l(1) = 1$ and $Q_u(0) = Q_l(0) = 0$. This classifier should now be a classifier based on fuzzy rough set theory that can handle noise. However, the next theorem shows that VQNN is the same classifier as FNN, provided the same similarity relation is used.

Theorem 2: Assume VQNN and FNN use the same similarity relation. Then VQNN and FNN return the same class for each target instance t.

Proof: As Q_u and Q_l are increasing, VQNN maximizes the following sum (the denominators in the arguments of Q_u and Q_l are equal):

$$\sum_{x \in NN} \min(R(x,t), C(x)).$$

As $C(x)$ only takes values 0 and 1, this sum can be written as:

$$\sum_{x \in NN} R(x,t)C(x),$$

which is exactly the sum FNN is maximizing over. ■

III. Positive Region Based Fuzzy Rough Nearest Neighbour Classification (POSNN)

From the previous section, we conclude that neither FRNN, nor VQNN is able to use concepts from fuzzy rough set theory to improve FNN. In the first case, we proved that only one instance is used for the classification, and in the second case, we obtain the same classifier as FNN, provided we use the same similarity measure.

In this section, we propose a new fuzzy rough based nearest neighbour classifier. The FNN classifier makes no assumptions on the quality of the neighbours, that is, each neighbour is considered equally important. However, there are three types of neighbours that may be less suitable for guiding the FNN classification:

1) *Border neighbours*: if the target instance has a neighbour that is on the border of two classes, this neighbour is less interesting than neighbours that are more at the center of a class.
2) *Instances in overlapping regions of classes*: if the target instance has a neighbour that is in the overlap of two classes, this neighbour is less interesting thanother neighbours. This case is closely related to the previous type of neighbours.
3) *Mislabeled neighbours*: if there is a neighbour that is isolated, that is, there are no instances of the same class lying close to it, this neighbour is probably noisy (mislabeled).

Fuzzy rough sets were designed to model this imperfect knowledge. The so-called positive region expresses for each instance to which extent instances similar to it are in the same class. For classification problems, it is given by the following formula:

$$POS(x) = \min_{y \in X} \mathcal{I}(R(x,y), R_c(x,y))$$

Here, $R_c(x,y)$ is 1 if y and x belong to the same class, and 0 otherwise. The similarity measure R is the one defined in Equation (2). Border instances or instances in overlapping regions will have smaller positive region memberships than instances in the center of a class, and mislabeled instances will have a lower positive region membership than instances that are surrounded by instances of the same class. This makes the positive region suitable to measure the quality of a neighbour.

The final fuzzy rough positive region based nearest neighbour algorithm (POSNN) proceeds as follows to classify an instance t:

- Determine NN, the set of K nearest neighbours of t
- Assign t to the class C for which

$$\frac{\sum\limits_{x \in NN} R(x,t)C(x)POS(x)}{\sum\limits_{x \in NN} R(x,t)}$$

is maximal.

Here, C can be $C_{gradual}$ or C_{crisp}, $R(x,t)$ is the similarity measure as defined in Equation (1). The use of the different similarity measures might be confusing, but we opt to use the similarity measure defined by Keller et al. in [2] to determine the similarity between the neighbours and the target instance, and the similarity measure in [6], which is typically used in fuzzy rough set theory, for calculating the positive region.

The difference between FNN and POSNN is that POSNN not only rewards neighbours that are closer to the instance, but also takes into account the quality (positive region) of the neighbours.

In the next section, we experimentally evaluate the POSNN classifier.

IV. Experimental Evaluation

A. Experimental set-up

We apply the POSNN classifier to 40 datasets from the UCI repository [10]. Their main characteristics are listed in Table I. We selected datasets that cover both nominal and/or continuous features, and vary in the number of features and instances they contain. In order to keep the running time of the experiments under control, we limited the maximum number of instances to 4000.

The POSNN classifier is compared to the KNN, FNN, and FRNN classifier. As it was shown in Section II-D that VQNN returns the same results as FNN, we do not explicitly list its results here. All methods are implemented in the Weka software platform [11]. For POSNN and FNN, we both apply the $C_{gradual}$ and C_{crisp} class membership functions. As a result, there are 6 classifiers: KNN and FRNN, FNN and POSNN applied with C_{crisp}, and FNN and POSNN applied with $C_{gradual}$.

For each experiment, we follow a 2×10 fold cross validation strategy: we divide the data in 10 folds and classify the instances of each fold using the remaining folds as training data. We repeat this 2 times and average the results.
To test the robustness of POSNN, we vary the parameter K representing the number of nearest neighbours from 1 to 15.

B. Results

In Figure 2 and Figure 3, we plot the average classification accuracies over all datasets of KNN, FNN, FRNN and POSNN. In Figure 2 we use the $C_{gradual}$ measure for FNN and POSNN, and in Figure 3 we use C_{crisp}. These results can also be found in Table II. In Table III, we show the best classification accuracy over all values K=$\in \{1,\ldots,15\}$ for each dataset.
From these plots and tables, we see that KNN and FRNN perform worse than FNN and POSNN. Moreover, KNN and FRNN strongly depend on the parameter K, while FNN and POSNN are more or less independent of K, assuming that K is at least 5. The optimal value K for POSNN and FNN is 6. The results for FNN and POSNN are similar, but the POSNN accuracies are consequently higher than FNN, except for those where K is equal to 4 or 14 as can be seen in Table II. The results using the C_{crisp} class membership degree are slightly better than those using $C_{gradual}$, and the differences are larger for larger K. The optimal K value is 5 or 6 for all considered classifiers.

TABLE I
Characteristics of the datasets: number of classes, attributes, instances, nominal attributes and continuous attributes.

Dataset	# classes	# atts	# inst	# nom	# cont
Water 2	2	38	390	0	38
Water 3	3	38	390	0	38
Cleveland	5	13	297	0	13
Glass	6	9	214	0	9
Heart	2	13	270	0	13
Letter	4	16	3113	0	16
Olitos	4	25	120	0	25
Wine	3	13	178	0	13
Anneal	5	38	898	32	6
Anneal-o	5	38	898	32	6
Audiology	24	69	226	69	0
Autos	6	25	205	10	15
Balance S	3	4	625	0	4
Breast	2	9	286	9	0
Wisconsin	2	9	699	0	9
Horse-c	2	22	368	15	7
Horse-co	2	27	368	20	7
Credit	2	15	690	9	6
German	2	20	1000	13	7
Pima	2	8	768	0	8
Cleveland-14	2	13	303	7	6
Hungarian-14	2	13	294	7	6
Heart S	2	13	270	0	13
Hepatitis	2	19	155	13	6
Hypothyroid	4	29	3772	22	7
Ionosphere	2	34	351	0	34
Iris	3	4	150	0	4
Kr-vs-kp	2	36	3196	36	0
Labor	3	16	57	8	8
Lymphograpy	4	18	148	15	3
Primary	3	17	339	17	0
Segment	7	19	2310	0	19
Sick	2	29	3772	22	7
Sonar	2	60	208	0	60
Soybean	19	35	683	35	0
Splice	3	61	3190	61	0
Vehicle	4	18	846	0	18
Vote	2	16	435	16	0
Vowel	11	13	990	3	10
Zoo	7	17	101	16	1

V. Conclusion and Future Work

In this paper we presented a new fuzzy nearest neighbour classifier, POSNN. It extends the FNN classifier by taking into account the quality of the nearest neighbours by means of the fuzzy positive region. We proved that previous attempts to incorporate fuzzy rough set theory in the FNN classifier are not able to use the main ingredients of fuzzy rough set theory: FRNN only uses information about one of the nearest neighbours, and VQNN is the same classifier as FNN if it uses the same similarity measure.

A preliminary experimental study showed that POSNN performs better than KNN and FRNN. The differences between POSNN and FNN are not large, but POSNN performs slightly better than FNN for almost all values K between 1 and 15.
The POSNN classifier can still be improved. One option is to use ordered weighted average rough sets [12] to define

TABLE II
AVERAGE CLASSIFICATION ACCURACIES OF KNN, FRNN, FNN AND POSNN FOR DIFFERENT VALUES OF K

K	KNN	FRNN	FNN $C_{gradual}$	POSNN $C_{gradual}$	FNN C_{crisp}	POSNN C_{crisp}
1	79.60	79.60	79.60	79.60	79.60	79.60
2	77.18	80.06	80.29	80.86	80.29	80.66
3	81.09	81.14	81.21	81.28	81.36	81.47
4	79.88	81.43	81.62	81.25	81.62	81.67
5	81.53	81.35	81.84	81.87	81.86	81.94
6	80.56	81.24	81.86	81.87	81.93	81.96
7	80.74	80.66	81.75	81.80	81.84	81.93
8	80.01	80.33	81.71	81.79	81.78	81.87
9	79.81	79.81	81.46	81.48	81.73	81.82
10	79.20	79.45	81.43	81.45	81.63	81.76
11	79.04	78.99	81.43	81.44	81.78	81.82
12	78.66	78.89	81.34	81.43	81.71	81.80
13	78.64	78.81	81.33	81.38	81.66	81.71
14	78.10	78.70	81.31	81.32	81.70	81.67
15	78.11	78.19	81.06	81.13	81.58	81.64

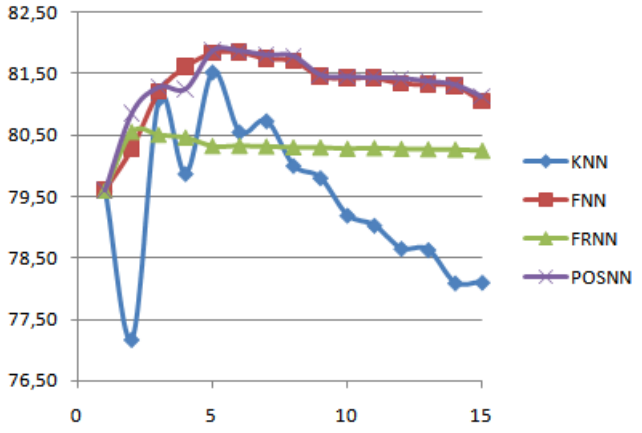

Fig. 2. Classification results for a range of K values, using $C_{gradual}$ for the class membership function.

the fuzzy positive region in order to deal with noise. Another option is to use POSNN for regression problems.

ACKNOWLEDGMENT

The paper has been partially supported by spanish projects TIN2011-28488 and TIC-2010-06858.

REFERENCES

[1] T. Cover and P. Hart, "Nearest neighbor pattern classification," *IEEE Transactions on Information Theory*, vol. 13, no. 1, pp. 21–27, 1967.
[2] J. M. Keller, M. R. Gray, and J. R. Givens, "A fuzzy k-nearest neighbor algorithm," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 15, pp. 580–585, 1985.
[3] L. A. Zadeh, "Fuzzy sets," *Information and Control*, vol. 8, no. 3, pp. 338–353, 1965.
[4] M. Sarkar, "Fuzzy-rough nearest neighbor algorithms in classification," *Fuzzy Sets and Systems*, vol. 158, no. 19, pp. 2134 – 2152, 2007.

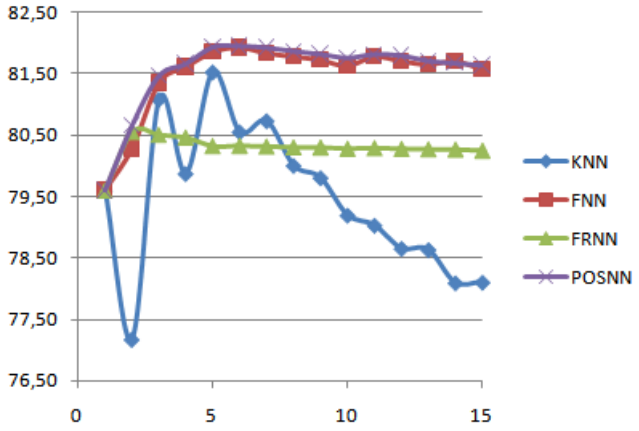

Fig. 3. Classification results for a range of K values, using C_{crisp} for the class membership function.

[5] R. Jensen and C. Cornelis, "A new approach to fuzzy-rough nearest neighbour classification," in *Proceedings of the 6th International Conference on Rough Sets and Current Trends in Computing*, ser. RSCTC '08, 2008, pp. 310–319.
[6] ——, "Fuzzy-rough nearest neighbour classification and prediction," *Theoretical Computer Science*, vol. 412, pp. 5871–5884, 2011.
[7] C. Cornelis, M. De Cock, and A. M. Radzikowska, "Vaguely quantified rough sets," in *Proceedings of the 11th International Conference on Rough Sets, Fuzzy Sets, Data Mining and Granular Computing*, 2007, pp. 87–94.
[8] R. O. Duda, P. E. Hart, and D. G. Stork, *Pattern Classification*. Wiley, 2001.
[9] C. Cornelis, M. De Cock, and A. M. Radzikowska, "Fuzzy rough sets: From theory into practice," in *Handbook of Granular Computing*, 2008, pp. 533–552.
[10] D. N. A. Asuncion, "UCI machine learning repository," 2007. [Online]. Available: http://www.ics.uci.edu/∼mlearn/MLRepository.html
[11] I. H. Witten and E. Frank, *Data Mining: Practical Machine Learning Tools and Techniques*, 2nd ed., ser. Morgan Kaufmann Series in Data Management Systems, 2005.
[12] R. Yager, "On ordered weighted averaging aggregation operators in multicriteria decisionmaking," *IEEE Transactions on Systems, Man and Cybernetics*, vol. 18, no. 1, pp. 183 –190, 1988.

TABLE III
BEST CLASSIFICATION ACCURACY OVER ALL VALUES K$\in \{1, \ldots, 15\}$ FOR EACH DATASET.

	KNN	FRNN	FNN $C_{Gradual}$	POSNN $C_{Gradual}$	FNN C_{Crisp}	POSNN C_{Crisp}
Water 2	86.54	83.97	85.51	85.38	85.64	85.26
Water 3	84.62	81.79	83.59	83.72	83.85	84.23
Cleveland	58.59	54.90	58.60	58.59	57.91	57.42
Glass	69.13	71.02	71.70	71.46	71.72	71.92
Heart	83.89	77.96	83.70	84.26	83.15	83.15
Letter	95.41	95.36	95.74	95.70	95.90	95.81
Olitos	82.92	76.25	82.08	82.92	82.08	82.92
Wine	96.91	95.77	97.21	97.78	97.21	97.78
Anneal	99.11	99.11	99.11	99.11	99.11	99.11
Anneal O	89.87	90.09	90.20	90.20	90.31	90.31
Audiology	72.98	76.12	75.67	75.67	75.67	75.67
Autos	77.00	78.48	78.26	78.05	77.75	77.04
Balance S	89.35	79.11	89.67	89.67	89.51	89.51
Breast	74.16	71.57	73.66	73.84	73.47	74.01
Wisconsin	97.00	94.49	96.93	96.93	96.93	97.00
Horse C	82.34	76.09	77.98	77.98	77.31	77.31
Horse CO	66.85	67.66	67.26	67.26	68.21	68.21
Credit	86.59	79.78	86.52	86.59	85.43	85.58
German	74.60	72.10	74.80	74.80	75.35	75.35
Pima	74.93	70.76	75.19	75.13	74.22	75.13
Cleveland 14	82.83	76.74	83.48	83.32	82.17	82.50
Hungarian 14	82.84	76.67	82.68	82.68	80.80	80.80
Heart S	80.56	77.96	80.93	80.56	81.11	81.67
Hepatitis	86.77	79.75	82.94	82.94	82.94	82.94
Hypothyroid	93.53	92.09	93.52	93.41	93.46	93.09
Ionosphere	86.77	91.31	86.77	87.48	86.77	87.61
Iris	96.67	96.00	96.00	96.00	96.00	96.00
Kr-vs-kp	96.25	90.55	96.35	96.35	96.64	96.64
Labor	81.67	85.33	81.67	81.67	84.00	84.00
Lymphograpy	85.48	81.05	85.45	85.45	85.45	85.45
Primary	43.06	34.94	42.92	42.77	42.92	42.77
Segment	97.08	97.08	97.08	97.08	97.08	97.08
Sick	96.43	96.14	96.39	96.42	96.42	96.30
Sonar	85.88	86.83	85.88	87.79	85.88	87.79
Soybean	56.64	64.92	63.74	63.60	63.74	63.60
Splice	87.02	75.88	85.89	85.89	85.94	85.94
Vehicle	71.81	70.10	72.75	72.16	73.22	72.51
Vote	94.04	83.22	84.96	84.96	82.99	82.99
Vowel	99.09	99.24	99.09	99.09	99.09	99.09
Zoo	96.59	97.14	96.59	96.59	96.59	96.59
AVERAGE:	83.60	81.13	83.46	83.53	83.35	83.45