Marcin PLUCIŃSKI

West Pomeranian University of Technology, Faculty of Computer Science and Information Technology

Application of the information-gap theory for evaluation of nearest neighbours method robustness to data uncertainty

***Abstract**. The paper describes a new method based on the information-gap theory which enables an evaluation of worst case error predictions of the kNN method in the presence of a specified level of uncertainty in the data. There are presented concepts of a robustness and an opportunity of the kNN model and calculations of these concepts were performed for a simple 1-D data set and next, for a more complicated 6-D data set. In both cases the method worked correctly and enabled evaluation of the robustness and the opportunity for a given lowest acceptable quality r_c or a windfall quality r_w. The method enabled also choosing of the most robust kNN model for a given level of an uncertainty α.*

***Streszczenie.** W artykule opisane jest zastosowanie teorii luk informacyjnych do określania największego błędu modelu kNN w przypadku wystąpienia w danych niepewności o określonym poziomie. Przedstawione zostały pojęcia odporności i sposobności modelu kNN oraz pokazane zostały przykłady ich wyznaczania dla prostych danych jednowejściowych i bardziej złożonych, sześciowejściowych. W obu przypadkach metoda działała prawidłowo, a dodatkowo umożliwiała wyznaczanie najbardziej odpornego modelu kNN przy określonym poziomie niepewności α. (**Zastosowanie teorii luk informacyjnych do wyznaczania odporności metody najbliższych sąsiadów na niepewność danych**).*

Keywords: k-nearest neighbours method, local regression, function approximation, information-gap theory, data uncertainty.
Słowa kluczowe: metoda k najbliższych sąsiadów, regresja lokalna, aproksymacja funkcji, teoria luk informacyjnych, niepewność danych.

Introduction

k-nearest neighbours method (kNN) belongs to the, so called, memory based approximation methods. It is one of the most important between them and probably one of the best described in many versions [1,2,3], and what is significant, it is still the subject of new researches [4,5]. Other memory based methods can be exemplary: methods based on locally weighted learning [1,6] which use various ways of samples weighting. Methods widely applied in this category are also probabilistic neural networks and generalised regression networks [7,8].

Learning of function approximators with an application of memory-based learning methods is very often an attractive approach in comparison with creating of global models based on a parametric representation. In some situations (for example: small number of samples), building of global models can be difficult and then memory-based methods become one of possible solutions for the approximation task.

k-nearest neighbours method

In the kNN method, during calculations of an answer for a question point $\mathbf{x}^*$ only k nearest (in a meaning of an applied metric – here Euclidean metric) samples are taken into account. In the classic kNN method, the model answer is calculated as a mean value of target function values or a weighted mean value. In such case, weight values usually depend on a distance $\delta(\mathbf{x}^*,\mathbf{x})$ between the question point $\mathbf{x}^*$ and analysed neighbours $\mathbf{x}$, for example:

$$w_{x^*,x} = \frac{1}{1 + m \cdot \delta(\mathbf{x}^*,\mathbf{x}) / k^2}, \tag{1}$$

where: the m parameter is taken empirically. The kNN method realises a local regression and the answer for the considered question point is calculated on the base of a local model created for k-nearest neighbours.

The main parameter of the kNN method is the number of neighbours k that are taken into account. It can be constant for entire data set, but in some approaches it can be dynamically varied – according to the question point location in the input space.

One of popular techniques of k evaluation is applying 'leave one out' crossvalidation or applying two distinct data sets: training data – that are memorised by the model, and testing data – to evaluate the real model error. The best k value is the value that gives the lowest test or crossvalidation error. Typical plot of a crossvalidation error as a function of neighbours number k taken into account is presented in Fig. 1b.

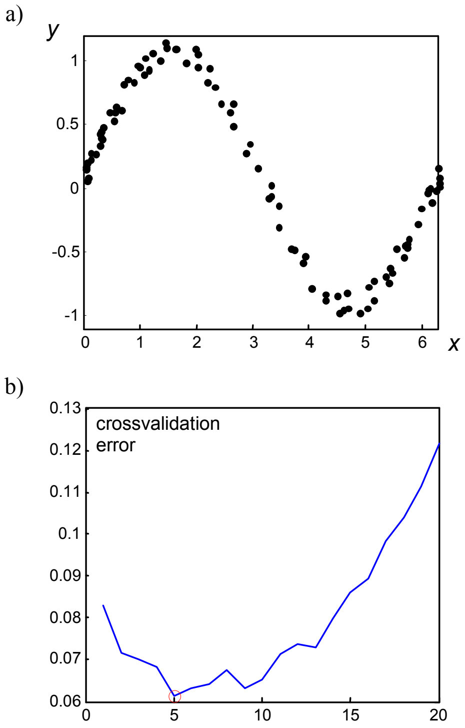

Fig.1. Exemplary data (a) and plot of a crossvalidation error as a function of number of neighbours k (b)

The lowest test or crossvalidation error guarantees the lowest real error of the model and the best generalisation.

Model of uncertainty

There are a lot of techniques that allow taking data uncertainty (both input and output one) into consideration. Among them we can mention methods that analyse data from a probabilistic point of view [1,2,9]. Other methods apply the theory of fuzzy sets [10,11]. In this paper there will be applied the information-gap model of uncertainty based on the theory described in [12,13,14,15] and on the interval analysis [16].

The information-gap theory enables a prediction of bounded worst case errors in the presence of a specified level of the uncertainty in the input and output data. The method also enables a discrimination between various models if there's need to find the model that is the most robust to the data uncertainty.

Interval analysis

In the paper there will be analysed data that were made uncertain by applying an interval expansion of size α in all dimensions of the data set (both in input and output attributes) [16]. The α parameter describes the unknown horizon of uncertainty in the information-gap model (described in the next subsection). After expansion each attribute becomes an interval number [16] and can be defined as an ordered pair of real numbers $[a,b]$ with $a < b$ such that:

(2) $$[a,b] = \{x : a \le x \le b\} \ .$$

During expansion each crisp attribute x is replaced by the interval $[\underline{x},\overline{x}]$ where $\underline{x}$ represents the lower interval bound and $\overline{x}$ the upper interval bound:

(3) $$[\underline{x},\overline{x}] = [x-\alpha, x+\alpha] \ .$$

The special interval number arithmetic is described in details in [16]. With its application it is possible to apply kNN method absolutely without changing the main algorithm of the method. Next section describes some experiments where Matlab toolbox INTLAB (created by S.M. Rump and described in [17]) was applied for calculations on interval numbers. The kNN model created with the application of interval arithmetic works correctly both for interval data and crisp data.

Information-gap model of uncertainty

Data gathered by miscellaneous measurement equipment usually are burdened with a certain error. Such data can be stored in the form of interval numbers for which there exists a possibility of mathematical calculations.

Other approach in modelling of the uncertainty surrounding each data vector $\mathbf{x}_i$, $i = 1 \ldots M$, is representing it by defining a local information-gap model [14,15]:

(4) $$L(\alpha,\mathbf{x}_i) = \left\{\mathbf{x} \in \mathbf{R}^N : \left\|\mathbf{x}-\mathbf{x}_i\right\|_\infty \le \alpha\right\} \ , \quad \forall \alpha \in \mathbf{R},$$

where: $\left\|\ldots\right\|_\infty$ is the infinity norm in $\mathbf{R}^N$ defined as:

$$\left\|\mathbf{x}\right\|_\infty = \max_j \left|x_j\right|$$

where: x_j is the j-th attribute of the vector $\mathbf{x}$ ($j = 1 \ldots N$) and $\alpha > 0$ is the unknown horizon of uncertainty. $\mathrm{L}(\alpha,\mathbf{x}_i)$ can be treated as an unbounded family of hypercuboid sets (for infinity norm) of possible $\mathbf{x}_i$ realisations.

Let's assume that we have 2 normalised distinct data sets. First of them will contain training data (although in the case of memory methods like the kNN method there is no real training process) and each data sample will consist of the input vector $\mathbf{x}_k$ and the target output value y_k, $k = 1 \ldots L$. The second data set will contain testing data and each data sample will also consist of the input vector $\mathbf{x}_i$ and the target output value y_i, $i = 1 \ldots M$.

For the crisp data, an error of modelling for the single testing sample $\mathbf{x}_i$ can be evaluated exemplary as:

$$\delta_i = \left|y_i - y_i^*\right| \ ,$$

where: y_i^* is the model answer for the question point $\mathbf{x}_i$ and a mean absolute error for the entire testing data set can be calculated as:

(5) $$e_{MAE} = \frac{1}{M}\sum_{i=1}^{M}\left|y_i - y_i^*\right| \ .$$

Now, we must take into account that data are uncertain so an accuracy of the model should be evaluated in a different way. The data are interval values so the accuracy will be also the interval value and its lower interval bound will be equal to a robustness function and an upper interval bound will be equal to an opportunity function [13]. Concepts of such functions are introduced by the information-gap theory and are described later.

Both y_i and y_i^* will be interval numbers so their subtraction result can be evaluated as:

$$d_i = y_i - y_i^* = \left[\underline{y_i} - \overline{y_i^*}, \overline{y_i} - \underline{y_i^*}\right] \ ,$$

according to the interval arithmetic [16]. The model error will be interval number:

(6) $$\delta_i = \left[\underline{\delta_i}, \overline{\delta_i}\right] \ ,$$

where the lower bound can be calculated as:

(7) $$\underline{\delta_i} = \begin{cases} 0 & \text{if } y_i \cap y_i^* \ne \varnothing \\ \min\{|\underline{d_i}|,|\overline{d_i}|\} & \text{otherwise,} \end{cases}$$

and the upper bound as:

(8) $$\overline{\delta_i} = \max\{|\underline{d_i}|,|\overline{d_i}|\} \ .$$

Fig. 2 illustrates the way of calculating lower and upper bound of the model error.

Now we can evaluate an interval value for the mean absolute error of the entire testing set as:

$$e_{MAE} = \left[\underline{e_{MAE}}, \overline{e_{MAE}}\right] \ ,$$

where:

$$\underline{e_{MAE}} = \frac{1}{M}\sum_{i=1}^{M}\underline{\delta_i} \quad \text{and} \quad \overline{e_{MAE}} = \frac{1}{M}\sum_{i=1}^{M}\overline{\delta_i} \ .$$

Next, additionally let's define the notion of the model accuracy as:

(9) $$q = \frac{1}{1+e_{MAE}} = \left[\underline{q},\overline{q}\right] = \left[\frac{1}{1+\overline{e_{MAE}}}, \frac{1}{1+\underline{e_{MAE}}}\right] .$$

The accuracy defined in such way changes its value in the range from 0 (for model errors approaching infinity) to 1 (for model errors approaching 0). It is clear that always $\underline{q} \le \overline{q}$.

Now, let's define an information-gap robustness function as equal to $\underline{q}$. Let r_c be the lowest acceptable model quality. The robustness of the model is the greatest horizon of uncertainty at which the model quality lower bound is equal or greater than r_c:

(10) $$\hat{\alpha}(r_c) = \max\left\{\alpha : \underline{q} \ge r_c\right\} \ .$$

The $\underline{q}$ value decreases with increasing α (Fig. 4) so the robustness increases with decreasing r_c:

$$r_c < r_c' \quad \Rightarrow \quad \hat{\alpha}(r_c) \ge \hat{\alpha}(r_c') \, .$$

Next, we can define notions of the opportunity function equal to $\overline{q}$ and the opportuneness of the model as the lowest horizon of uncertainty at which the model quality upper bound is equal or greater than r_w value (windfall quality):

(11) $$\hat{\beta}(r_w) = \min\left\{\alpha : \overline{q} \ge r_w\right\} \, .$$

$\hat{\beta}(r_w)$ is increasing as r_w is getting larger (Fig. 4):

$$r_w > r_w' \quad \Rightarrow \quad \hat{\beta}(r_w) \ge \hat{\beta}(r_w') \, ,$$

and it means that increasing the windfall quality of the model r_w causes an increase in the level of uncertainty $\hat{\beta}(r_w)$ needed to obtain that windfall.

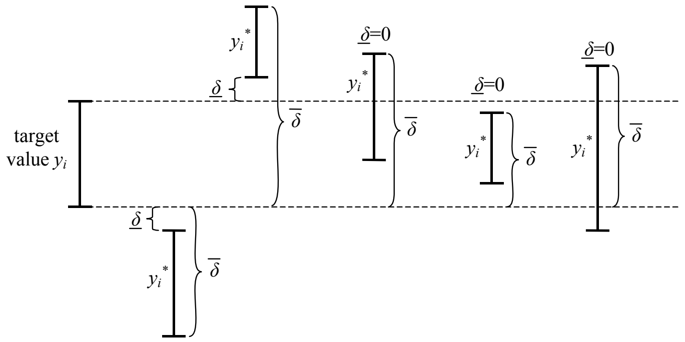

Fig.2. Exemplary calculations of the lower and upper bound of the model error

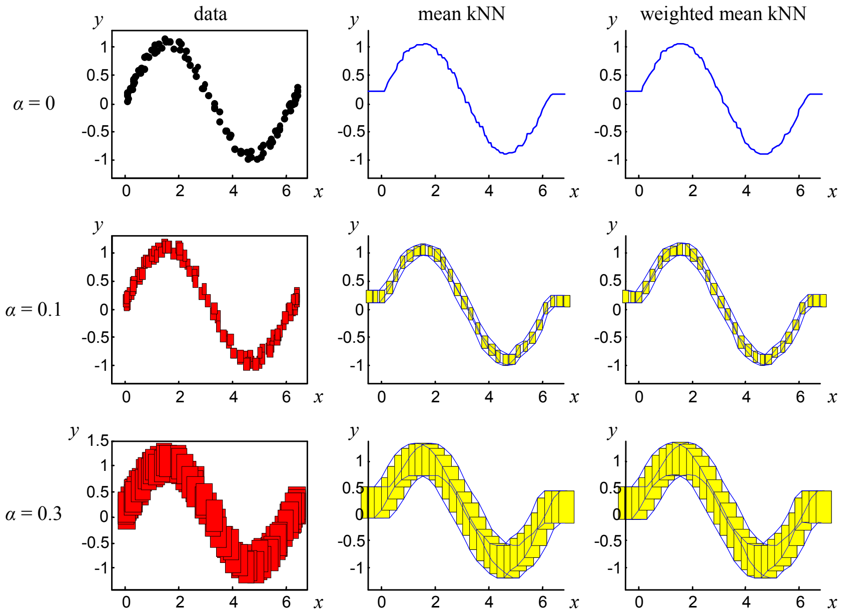

Fig.3. Exemplary data and model characteristics for various α values

Experiments

In the beginning let's find out how the kNN method works with uncertain data. The crisp data were submitted to the interval expansion of the size α in all dimensions of the data set (in the way described in the previous section). Fig. 3 presents data and model characteristics for various α values. The kNN model works correctly and it can be observed that the model characteristic width increases with increasing of the α value.

Now, let's take under consideration the model based on the crisp data from Fig. 1a. The number of nearest neighbours k is set to 5, and the α value will grow from 0 to 0.1. We can evaluate the lower and the upper bound of the model quality from equation (9), Fig. 4.

Fig. 4 can be used to quantify the performance under uncertainty of a given model and directly access the robustness of the model at any demanded quality. For example, by setting r_c to 0.9, an error of up to 0.024 can be tolerated in all elements of the measured vectors $\mathbf{x}_i$, without the risk of decreasing quality below 0.9. That is, the robustness is $\hat{\alpha}(r_c = 0.9)$ = 0.024. Additionally, such level of uncertainty provides the opportunity for the model quality of up to 0.978, (opportunity function is $\hat{\beta}(r_w = 0.978)$ = 0.024).

The model robustness notion can also be applied for a discrimination between various kNN models. For a certain level of uncertainty α, the best will be the model with the greatest value of $\hat{\alpha}$ and the lowest value of $\hat{\beta}$. Fig. 5 presents the plot of robustness function values for kNN models with various nearest neighbour number k and the level of uncertainty α set to 0.03. From the figure we can see that the most robust (for the uncertainty α = 0.03) is the kNN model with k = 5.

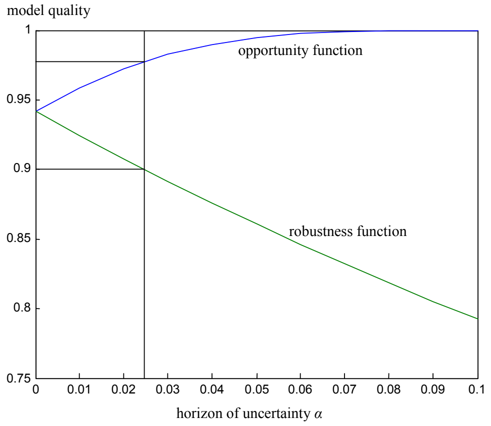

Fig.4. The lower and the upper bound of the model quality for various α values

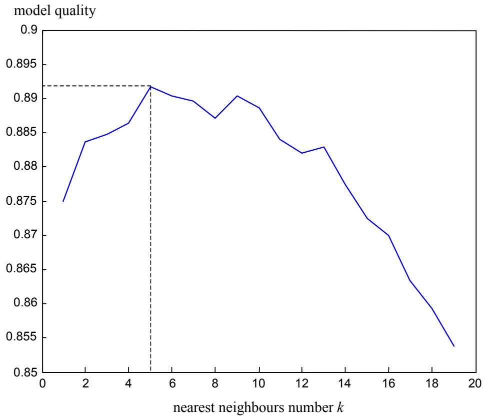

Fig.5. The plot of robustness function values for kNN models with various nearest neighbours number k and α = 0.03

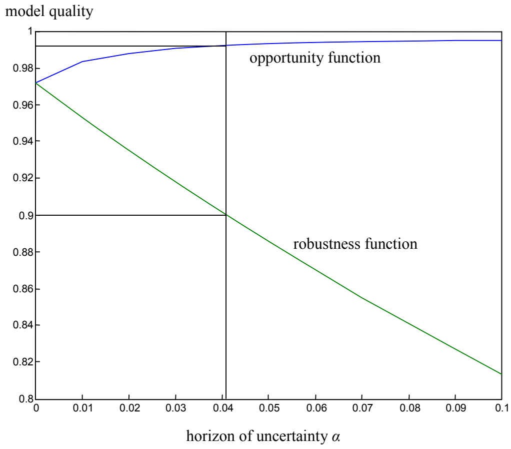

Fig.6. The lower and the upper bound of the model quality for various α values (figure created for 'cpu' data)

Of course, the described method for an evaluation of the model robustness can also be applied for a more complicated data with a greater number of input attributes. Exemplary calculations for the popular benchmark data 'cpu' are presented below. (The benchmark can be found in UCI Machine Learning Repository: http://archive.ics.uci.edu/ml/datasets.html.) Data have 6 input attributes, so the first performed step was a normalisation. Next, the number k was set to 5 and α value was changed from 0 to 0.1. Fig. 6 presents the robustness function and the opportunity function for the 'cpu' data. If we set r_c to 0.9 we can find the model robustness $\hat{\alpha}(r_c = 0.9)$ = 0.041. As before, it means that an error of up to 0.041 will not decrease quality below 0.9.

Conclusions

Errors are a natural property of data, so if uncertain data are used for a model creation it is important to evaluate the robustness to uncertainty of such models.

The paper has described an approach based on the information-gap theory which enables an evaluation of worst case error predictions of the kNN method in the presence of a specified level of uncertainty in the data. There were presented concepts of the robustness and the opportunity of the kNN model and calculations of these concepts were performed for the simple 1-D data set and next, for the more complicated 6-D data set. In both cases (and also in many other experiments realised by the author) the method worked correctly and enabled evaluation of the robustness and the opportunity for the given lowest acceptable quality r_c or the windfall quality r_w. The method enabled also choosing of the best (the most robust) kNN model for the given level of uncertainty α.

REFERENCES

[1] Cichosz P., Learning systems. WNT Publishing House, Warsaw, 2000, [in Polish]

[2] Hand D., Mannila H., Smyth P., Principles of data mining. The MIT Press, 2001

[3] Moore A.W., Atkeson C.G., Schaal S.A., Memory-based learning for control. *Technical Report CMU-RI-TR-95-18*, Carnegie-Mellon University, Robotics Institute, 1995

[4] Kordos M., Blachnik M., Strzempa D., Do we need whatever more than k-NN? *Proceedings of 10th International Conference on Artificial Intelligence and Soft Computing*, Zakopane, Poland, 414-421, Springer, 2010

[5] Korzeń M., Klęsk P., Sets of approximating functions with finite Vapnik-Czervonenkis dimension for nearest-neighbours algorithm. *Pattern Recognition Letters*, 32, 1882-1893, 2011

[6] Atkeson C.G., Moore A.W., Schaal S.A., Locally weighted learning. *Artificial Intelligence Rev.*, 11, 11-73, 1997

[7] Pluciński M., Application of data with missing attributes in the probability RBF neural network learning and classification. *Artificial Intelligence and Security in Computing Systems: 9th International Conference ACS'2002: Proceedings*, Eds.: J. Sołdek, L. Drobiazgiewicz, Boston/Dordrecht/London: Kluwer Academic Publishers, 63-72, 2003

[8] Wasserman P.D., Advanced methods in neural computing. New York, Van Nostrand Reinhold, 1993

[9] Wright W. A., Bayesian approach to neural network modeling with input uncertainty. *IEEE Transactions on Neural Networks*, vol. 10, no. 6, 1261–1270, 1999

[10] Duch W., Uncertainty of data, fuzzy membership functions, and multilayer perceptrons. *IEEE Transactions on Neural Networks*, vol. 16, no. 1, 10–23, 2005

[11] Piegat A., Fuzzy modeling and control. Physica Verlag, Heidelberg-New York, 2001

[12] Ben-Haim Y., Set-models of information-gap uncertainty: axioms and an inference scheme. *Journal of the Franklin Institute*, 336, 1093-1117, 1999

[13] Ben-Haim Y., Information-gap decision theory: decisions under severe uncertainty. New York: Academic Press, 2001

[14] Ben-Haim Y., Uncertainty, probability and information-gaps. *Reliability Engineering and System Safety*, 85, 249-266, 2004

[15] Pierce S.G., Ben-Haim Y., Worden K., Manson G., Evaluation of neural network robust reliability using information-gap theory. *IEEE Transactions on Neural Networks*, vol. 17, no. 6, 1349-1361, 2006

[16] Moore R. M., Kearfott R.B., Cloud M.J., Introduction to interval analysis, Society for Industrial and Applied Mathematics, Philadelphia, 2009

[17] Rump S.M., INTLAB – INTerval LABoratory. *Developements in Reliable Computing*, pp. 77-104, Kluwer Academic Publishing, Dordrecht, 1999

***Author**: dr inż. Marcin Pluciński, West Pomeranian University of Technology, Faculty of Computer Science and Information Technology, Żołnierska 49, 71-210 Szczecin, Poland, E-mail: mplucinski@wi.zut.edu.pl*