Библиографические ссылки

1. Воронов А. А., Титов В. К., Новогранов Б. Н. Основы теории автоматического регулирования. М. : Высш. шк., 1977.

2. Медведев А. В. Адаптация в условиях непараметрической неопределенности // Адаптивные системы и их приложения : сб. науч. тр. / Новосибирск : Наука, 1978. С. 4–34.

3. Медведев А. В. Непараметрические системы адаптации. Новосибирск : Наука, 1983.

4. Уткин В. И. Скользящие режимы и их применения в системах с переменной структурой. М. : Наука, 1974.

5. Kjaer M. A. Sliding Mode Control. Dept. of Autom. Control // Lund Institute of Technology, Sweden, 2004.

6. Slotine J.-J. E., Li W. Applied Nonlinear Control. Englewood Cliffs : Prentice-Hall, 1991.

7. Самарский А. А. Теория разностных схем. М. : Наука, 1989.

E. D. Agafonov

NONPARAMETRIC CONTROL ALGORITHM FOR NONLINEAR DYNAMIC SYSTEMS USING SLIDING MODES

The paper deals with a new control algorithm designed for nonlinear dynamic systems with the use of Sliding Mode control approach. The SISO object is represented by its nonparametric finite differences model in state space. The paper gives recommendations for tuning and optimization of the control algorithm. The proposed control algorithm is implemented in MATLAB/Simulink technical computing software. As an illustrative example we presrnt results of control process of inverted pendulum.

Keywords: nonparametric dynamic system, finite difference model, nonparametric control, Lypunov stability, sliding mode control.

K. Zablotskaya, S. Walter, S. Zablotskiy, W. Minker

DETERMINATION OF VARIABLES SIGNIFICANCE USING ESTIMATIONS OF THE FIRST-ORDER PARTIAL DERIVATIVE

In this paper we describe and investigate a method which allows us to detect the most informative features out of all data extracted from a certain data corpus. The significance of input features is estimated as an average absolute value of the first-order partial derivative. The method requires the values of the objective function at the certain assigned points. If there is no possibility to calculate these values (the object is not available for experiments), we use nonparametric kernel regression to approximate them. The algorithm is tested on different simulated objects and is used for investigation of the dependency between linguistic features of spoken utterances and speakers' capabilities.

Keywords: non-parametric kernel regression, first-order partial derivative.

In our research we try to investigate if there is a dependency between spoken utterances of a person and his capabilities. For this purpose we collected a corpus of monologues and dialogues of different speakers [1]. Their verbal intelligence was measured with an intelligence test [2]. Dealing with the corpus, we try to extract relevant information enough for clustering, classification, regression, or other data mining tasks. There are normally lots of different features which could be extracted from the monologues and dialogues, but their importance or relevance is not always obvious. Most of them are noise fields, which make the analysis of data increasingly difficult. When working with high dimensional spaces, the computational effort required by data analysis tools may be tremendous. It is therefore essential to detect ir-relevant or weakly correlated features and exclude them out of consideration.

There exist different solutions to this problem. One of them is the use of Pearson's coefficient or the coefficient of multiple correlation. However, if Pearson's coefficient is close to 0, it does not mean that the output and input variables are not correlated. It just shows that there is no linear dependency between them. Such features should not be excluded out of consideration without additional analysis. Another approach to decrease the number of features is Principal Component Analysis. This method involves a mathematical procedure that transforms correlated variables into a smaller number of uncorrelated ones called principal components. But it does not determine the contribution of a certain feature to an objective function.

In this paper we describe a method which determines the most informative features even if the dependency between input variables and the output is not linear.

Determination of the Most Informative Features. To determine if there is a dependency between input features (or extracted features) and the output, we make a series of experiments on the object (if it is available) or create a model using non-parametric kernel regression and estimate the average first-order partial derivative with respect to each input feature. The feature with the largest average partial derivative is the most important. This algorithm may be described in the following way.

Non-parametric kernel regression (NPR) allows us to create a model using the data set $x_1[t], ..., x_n[t]$, $y[t], t = \overline{1, s}$ without additional knowledge about the dependency structure [3; 4]. NPR estimates the dependency between inputs and outputs using a weighted average of the observations $y[t]$:

$$\hat{M}\{Y \mid x\} \equiv \hat{y}(\overline{x}) = \frac{\sum_{t=1}^{s} y[t] \prod_{i=1}^{n} \Phi\left(\dfrac{x_i - x_i[t]}{C_i}\right)}{\sum_{t=1}^{s} \prod_{i=1}^{n} \Phi\left(\dfrac{x_i - x_i[t]}{C_i}\right)},$$

where C_i – is the bandwidth or smoothing parameter, $\Phi(z)$ – is a kernel function.

The kernel function assigns weights for each observation. The weighted sum of $y[t]$ estimates the output at any point $\overline{x}$. The parameters C_i determine how many points from the training data set are used for calculating $\hat{y}(\overline{x})$. The observations which are nearer to $\overline{x}$ have larger weights and are more significant for $\hat{y}(\overline{x})$. If C_i are large, a lot of observations are taken into account and the model is not precise. These parameters should be trained on the existing data set, and C_i providing the smallest mean square error (MSE) are used for other investigations.

Let an object have an input vector $\overline{x} = (x_1, x_2, ..., x_n)$ and an output $y = f(\overline{x})$. A feature x_i is informative if its average influence on the output is significant, given the other $n-1$ features fixed. We estimate this significance as an average absolute value of the first-order partial derivative with respect to this variable.

Let the variables $\overline{x} = (x_1, x_2, ..., x_n)$ belong to the intervals $[a_1; b_1]$, $[a_2; b_2]$, …, $[a_n; b_n]$. We generate random values $\{\tilde{x}_1[1], ..., \tilde{x}_1[m], \tilde{x}_2[1], ..., \tilde{x}_2[m], ...\}$ in corresponding intervals, m is a predefined value. To get a precise estimation of the average first-order partial derivative, we generate these random values near one observation value $\overline{x}[l]$, $l = \overline{1, s}$:

$$\sum_{t=1}^{s} \prod_{i=1}^{n} \Phi\left(\frac{\tilde{x}_i[k] - x_i[t]}{C_i}\right) > 0,$$

for all $k = \overline{1, m}$. Then we fix the features $(x_2, ..., x_n)$ at some points, for example at $(\tilde{x}_2[1], \tilde{x}_3[1], ..., \tilde{x}_n[1])$. The

outputs of the goal function are estimated at the following points:

$$\hat{y}_1^+ = f(\tilde{x}_1[1] + h_1^+, \tilde{x}_2[1], ..., \tilde{x}_n[1]),$$
$$\hat{y}_1^- = f(\tilde{x}_1[1] - h_1^-, \tilde{x}_2[1], ..., \tilde{x}_n[1]),$$
$$\hat{y}_2^+ = f(\tilde{x}_1[2] + h_2^+, \tilde{x}_2[1], ..., \tilde{x}_n[1]),$$
$$\hat{y}_2^- = f(\tilde{x}_1[2] - h_2^-, \tilde{x}_2[1], ..., \tilde{x}_n[1]), ...,$$

where h_1^+ and h_1^- are random values from a small interval (for example, $h \in [0,01; 0,5]$).

The first component of the average first-order partial derivative with respect to x_1 is estimated as:

$$\hat{f}'_{x_1}[1] = \frac{\dfrac{|\hat{y}_1^+ - \hat{y}_1^-|}{h_1^+ - h_1^-} + \dfrac{|\hat{y}_2^+ - \hat{y}_2^-|}{h_2^+ - h_2^-} + ... + \dfrac{|\hat{y}_m^+ - \hat{y}_m^-|}{h_m^+ - h_m^-}}{m}.$$

Then the features $(x_2, ..., x_n)$ are fixed at other points, for example at $(\tilde{x}_2[2], \tilde{x}_3[1], ..., \tilde{x}_n[1])$, and the same procedure is repeated for x_1. The average absolute value of the partial derivative in the neighborhood of $\overline{x}[l]$ is estimated as:

$$\hat{f}'_{x_1}(\overline{x}[l]) = \frac{\hat{f}'_{x_1}[1] + \hat{f}'_{x_1}[2] + ... + \hat{f}'_{x_1}[M]}{M},$$

where M – the number of all possible combinations, $M = m^{(n-1)}$. As these random values have been generated in the neighborhood of one observation values, only small space is investigated. We generate $\{\tilde{x}_1[1], ..., \tilde{x}_1[m], \tilde{x}_2[1], ..., \tilde{x}_2[m], ...\}$ next to another observation point $\overline{x}[l']$ and find $\hat{f}'_{x_1}(\overline{x}[l'])$ in the same way. This procedure is repeated K times. The average absolute value of the partial derivative $\hat{f}'_{x_1}$ – is estimated as: $\hat{f}'_{x_1} = \sum_{l=1}^{K} \hat{f}'_{x_1}(\overline{x}[l]) \Big/ K$, where K is a predefined value.

In the same way the average absolute values of $\hat{f}'_{x_i}$, $i = \overline{1, n}$ are estimated.

Investigation of the Algorithm. In this section we show the results of the algorithm's work when the object is not available for experiments, i. e. there is only collected data. In the following experiments the average absolute value of the partial derivative is estimated with $M = 2$ and $K = 20$. The function for simulating the object is given by: $f(\overline{x}) = 5x_1 + 0,5x_2 - 10x_3 + 0,1x_4 + 2x_5$.

In our first experiment, the non-parametric regression model is trained using all the input variables ($C_i = [0,4; 1,7; 0,3; 1,8; 0,8]$, $MSE = 0,08$).

Then we take away the first feature x_1 from the data set. In this case the situation with incomplete data is simulated, but the most informative features should nevertheless be found ($C_i = [1,9; 0,3; 1,9; 0,9]$, $MSE = 0,39$). The results of the algorithm are shown in Table 1. As we can see, the algorithm was able to find the most important features in both cases.

Table 1

Results of the algorithm's work

Features	5 inputs are used		4 inputs are used	
	$\hat{f}'_x$ (Real ranks)	Algorithm's $\hat{f}'_x$ (Ranks)	$\hat{f}'_x$ (Real ranks)	Algorithm's $\hat{f}'_x$ (Ranks)
x_1	5,0 (2)	3,42 (2)	–	–
x_2	0,5 (4)	1,40 (4)	0,5 (3)	0,39 (3)
x_3	10,0 (1)	6,51 (1)	10,0 (1)	7,97 (1)
x_4	0,1 (5)	1,20 (5)	0,1 (4)	0,28 (4)
x_5	2,0 (3)	2,10 (3)	2,0 (2)	1,39 (2)

Table 2

Results of the algorithm's work

Features	5 inputs are used		8 inputs are used	
	$\hat{f}'_x$ (Real ranks)	Algorithm's $\hat{f}'_x$ (Ranks)	$\hat{f}'_x$ (Real ranks)	Algorithm's $\hat{f}'_x$ (Ranks)
x_1	3,83 (5)	3,15 (5)	3,83 (5)	2,71 (5)
x_2	4,23 (4)	3,29 (4)	4,23 (4)	3,24 (4)
x_3	4,38 (3)	3,50 (3)	4,38 (3)	4,38 (3)
x_4	7,05 (2)	4,24 (2)	7,05 (2)	4,43 (2)
x_5	30,0 (1)	31,48 (1)	30,0 (1)	26,23 (1)
x_6	–	–	0,027 (7)	0,43 (7)
x_7	–	–	0,021 (8)	0,07 (8)
x_8	–	–	0,06 (6)	0,74 (6)

Now let's use the following function for generating the input and output data: $f(\overline{x}) = 7\sin(x_1) + 6\cos(x_2) - 8\sin(x_3) - 10\cos(x_4) + 5x_5^2$. In this dependency there are no features which influence on the output is linear. This is a more complex situation for the algorithm. However, if the model is trained well ($C_i = [1,0; 0,6; 0,7; 0,5; 0,1]$, $MSE = 0,32$), the algorithm gives us good results (see Table 2).

Let's use the same function for simulating the data set and add three more features to the input variables. We simulate the situation when the data set is large and not all features influence on the output. These additional input features are: $x_6 = 0,05\sin(t)$, $x_7 = 0,03\cos(t)$, $x_8 = 0,01t^2$. The coefficients with these features are small, so that x_6, x_7 and x_8 are noises for the output. In this case we use all the features to train the model. The results are shown in Table 2. The algorithm could find the most informative and the least informative features ($C_i = [1,0; 0,6; 0,7; 0,5; 0,1; 1,9; 1,5; 1,4]$, $MSE = 0,32$).

Let's use for simulating the function:

$$f(\overline{x}) = 0,2\sin(2x_1) + 2\cos(8x_2) + 5\sin(x_3) +$$
$$+ 0,1x_4 + 0,5x_5 + x_6 + 2x_7 + 3x_8 + 4x_9 + 5x_{10},$$

generate the data set and take away the features x_1, x_2 and x_3. The results of the algorithm ($C_i = [1,5; 1,4; 1,0; 0,9; 0,5; 0,5; 0,5]$, $MSE = 0,3$) are given in Table 3.

Analyzing the results in the tables, we may say that the algorithm with the non-parametric model can find the most informative features. This method can be used for analyzing a high dimensional data set. It allows us to exclude the least informative features from consideration.

Table 3

Results of the algorithm's work

Features	$\hat{f}'_x$ (Real ranks)	Algorithm's $\hat{f}'_x$ (Ranks)
x_1	–	–
x_2	–	–
x_3	–	–
x_4	0,1 (7)	0,61 (7)
x_5	0,5 (6)	0,71 (6)
x_6	1 (5)	1,26 (5)
x_7	2 (4)	1,80 (4)
x_8	3 (3)	3,79 (3)
x_9	4 (2)	4,15 (2)
x_{10}	5 (1)	4,83 (1)

Experiments with the Corpus. We decided to analyze different features extracted from monologues of German native speakers using the algorithm described above. The corpus consists of transcribed descriptions of a short film by different candidates. German native speakers of different ages and educational levels were asked to watch a short film and to describe it with their own words. The film was about an experiment on how long people could be without sleep. The participants were also asked to take an intelligence test. The verbal part of the test consists of 6 subtests. The first subtest is «Information». With this sub-test the general knowledge is measured; 25 questions come from a particular culture. For example, «What is the capital of Russia?» Overall, 56 candidates were tested; 3 hours 30 minutes of audio data were collected.

To extract features from the monologues, all the words from the descriptions were compared with a special dictionary [5]. The dictionary consists of different words sorted by 64 categories. For example, the category «Articles» contains words die, das, der, ein, eine, einen, etc. Each word from the dictionary may refer to several

categories. For example, the word traurig (sad) refers to the categories «Affect», «Negative emotion» and «Sadness». We analyzed all the monologues, calculated the number of words for each category and divided them by the total amount of words in each monologue. By this way we got 64 characteristics of 56 monologues. Our task was to investigate the dependency between these 64 features and the results of the subtest «Information», and to find several informative features out of 64 characteristics.

We combined 4 or 5 features together, trained the non-parametric model and applied our method. As a result, the category «Affection» had the largest value of the first-order partial derivative and was estimated as a more informative feature. «Positive emotions» and «Negative emotions» are subcategories of «Affection» and are also relevant according to our algorithm. However, «Anger» and «Optimism» do not have large values of $\hat{f}'_{x_i}$. The category «Cognitive mechanism» is estimated as irrelevant, however, the category «Cause» which is a subcategory of «Cognitive mechanism» is more important.

Discussion and Future work. The goal of this work was to apply the method to the corpora. In each combination of the features the category of emotions was determined as the most informative feature. It means that there is a dependency between speaker's general knowledge and the amount of emotional words which he uses in his speech. We could not find any references describing this dependency. Only in LEAS [6] emotional intelligence is measured linguistically, however, the cor-

relation between them was not found. For our algorithm we used a small data set that also influenced the results. Also, these emotional words may be a subcategory of another category which was not analyzed. For example, they may create a group of frequently-used words, or they are formed from abstract words which show the level of intelligence in spoken utterances. This research and the results are preliminary; in our future work we are going to further investigate this phenomenon, to find other linguistic features which reflect verbal intelligence and to collect more data for more precise estimations.

References

1. Zablotskaya K., Walter S., Minker W. Speech data corpus for verbal intelligence estimation. In Proceedings of International Conference on Language Resources (LREC) 2010.

2. Wechsler D. Handanweisung zum Hamburg-Wechsler-Intelligenztest fuer Erwachsene (HAWIE). Separatdr., Bern; Stuttgart; Wien, Huber. 1982.

3. Nadaraya E. On estimating regression. Theory of Probability and its Applications. 1964. Vol. 10. P. 186–190.

4. Watson G. Smooth regression analysis. Sankhya – The Indian Journal of Statistics. 1964. Vol. 26. P. 359–372.

5. Computergestuetzte quantitative textanalyse / M. Wolf, A. B. Horn, M. R. Mehl et al. // Diagnostica, 54. 2008. Heft 2. P. 85–98.

6. Lane R. D, Schwartz G. E. Levels of emotional awareness: a cognitive-developmental theory and its application to psychopathology. Am J Psychiatry. 1987. P. 113–143.

S. Zablotskiy, T. Müller, W. Minker

ESTIMATION OF RADIO SIGNAL QUALITY DEGRADATION BY MEANS OF NEURAL NETWORK AND NON-PARAMETRIC REGRESSION MODEL

In this paper we present an approach which allows us to avoid expansive and time consuming subjective assessments of audio quality degradation caused by different nature distortions while transmitting and receiving of stereo audio signal through the radio channel. This approach is based on the basic version of PEAQ (Perceptual Evaluation of Audio Quality) originally developed mainly for audio codec estimation. The MOV (Model Output Variables) vector of the PEAQ method is mapped to the audio quality degradation scale using two different models: neural networks and non-parametric regression. The results of two independent approaches are compared.

Keywords: PEAQ, audio quality degradation, neural network, non-parametric regression.

The manufacturers of radio receivers and other radio equipment have to estimate the quality of the new product comparing to the existent equipment. Among other things the common listening comprehension of the perceived degradation with respect to the original (reference) audio signal has to be taken into consideration because humans with their own listening comprehension are supposed to be the main end-users of the designed equipment. Since

any high-quality reliable subjective assessments are very expensive and time consuming it is strongly desired to have a tool for automatic perceptual evaluation of the audio quality degradation. This is a fundamental idea behind the PEAQ method, as specified in ITU-R BS.1387 recommendation [1]. According to this recommendation the PEAQ measurement method is applicable to the most types of audio signal processing equipment, both digital