An Algorithm of Classification for Nonstationary Case

Piotr Kulczycki*, and Piotr Andrzej Kowalski*

Polish Academy of Sciences, Systems Research Institute
{kulczycki,pakowal}@ibspan.waw.pl

Abstract. The paper deals with the classification task, where patterns are nonstationary. The method ensures the minimum expected value of misclassifications and is independent of patterns' shapes. This procedure eliminates elements of patterns with insignificant or even negative influence on the results' accuracy. Appropriate modifications follow the classifier parameters, which increases the effectiveness of procedure adaptation for nonstationary patterns. The number of patterns is not methodologically limited in the presented concept.

Keywords: data analysis, classification, pattern nonstationarity, pattern size reduction, classifier adaptation.

1 Introduction

Classification constitutes one of the basic procedures of data analysis and exploration [Han and Kamber, 2001]. In most of the methods used today, one assumes stationarity (unchanged by time) of patterns characterizing particular classes. However, more and more often, as models have become more accurate, and investigated phenomena have become more complex [Kulczycki et al, 2007], in particular those in which new – with the most current being the most valuable – elements are continuously added to patterns, this assumption is successfully ignored.

The concept of the method for classification with nonstationary patterns proposed in this paper was conceived on the basis of the sensitivity method used in artificial neural networks. As a result of its operation, particular elements of patterns receive weights proportional to their significance for correct results. Elements of the smallest weights are eliminated. For the sake of the patterns' nonstationarity, their elements whose weights are currently small but increase successively are kept. In addition a procedure is proposed ensuring that an adaptation to changing conditions is obtained by correcting classifier parameters values. Its formula is based on the Bayes approach, providing a minimum of potential losses arising from incorrect classification. It is also possible to introduce preferences for those classes whose elements – due to potential nonsymmetrical conditioning of the task – especially should not be mistakenly assigned to others. The classifier was constructed applying the statistical kernel estimators methodology, thus freeing the above procedure from arbitrary assumptions

* Also: Cracow University of Technology, Department of Automatic Control and Information Technology .

F. Castro, A. Gelbukh, M.G. Mendoza (Eds.): MICAI 2013, Part II, LNAI 8266, pp. 301–313, 2013.

regarding patterns' shapes – their identification is an integral part of the algorithm presented here.

The first sections of this paper, i.e. 2-7, briefly describe mathematical apparatus and component procedures used in the main part – Section 8 – to synthesize of the algorithm for classification with nonstationary case investigated here. The numerical verification and comparison with the similarly conditioned support vector machine concept [Krasotkina et al, 2011] is the subject of Section 9, followed by final comments and remarks.

A preliminary version of this paper was presented in part as [Kulczycki and Kowalski, 2013a].

2 Statistical Kernel Estimators

Consider the n-dimensional random variable X, with a distribution characterized by the density f. Its kernel estimator $\hat{f}:\mathbb{R}^n \to [0,\infty)$ is calculated on the basis of the random sample

$$x_1,\ x_2, \ldots, x_m \quad , \tag{1}$$

and defined – in the basic form – by the formula

$$\hat{f}(x) = \frac{1}{mh^n} \sum_{i=1}^{m} K\left(\frac{x - x_i}{h} \right) \quad , \tag{2}$$

where the positive coefficient h is known as a smoothing parameter, while the measurable function $K:\mathbb{R}^n \to [0,\infty)$ symmetrical with respect to zero, having at this point a weak global maximum and fulfilling the condition $\int_{\mathbb{R}^n} K(x)\,\mathrm{d}x = 1$ is termed a kernel. The monographs [Kulczycki, 2005; Silverman, 1986; Wand and Jones, 1995] contain a detailed description of the above methodology.

In this paper the generalized (one-dimensional) Cauchy kernel is applied, in the multidimensional case generalized by the product kernel concept [Kulczycki, 2005 – Section 3.1.3; Wand and Jones, 1995 – Sections 2.7 and 4.5]. For calculation of the smoothing parameter, the simplified method assuming the normal distribution [Kulczycki, 2005 – Section 3.1.5; Wand and Jones, 1995 – Section 3.2.1] can be applied, thanks to the positive influence of this parameter correction procedure, presented below in Section 7. For general improvement of the kernel estimator quality the modification of the smoothing parameter [Kulczycki, 2005 – Section 3.1.6; Silverman, 1986 – Section 5.3.1] will be applied, with the intensity $c \geq 0$; as its initial standard value $c = 0.5$ can be assumed.

Details are found in the monographs [Kulczycki, 2005; Silverman, 1986; Wand and Jones, 1995].

3 Bayes Classification

Consider J sets consisting of elements of the space $\mathbb{R}^n$:

$$x_1^{'},\ x_2^{'},\ldots,x_{m_1}^{'} \tag{3}$$

$$x_1^{''},\ x_2^{''},\ldots,x_{m_2}^{''} \tag{4}$$

$$\vdots$$

$$x_1^{''\ldots'},\ x_2^{''\ldots'},\ldots,x_{m_J}^{''\ldots'}\ , \tag{5}$$

representing assumed classes. The sizes m_1, $m_2,\ldots,m_J$ should be proportional to the "contribution" of particular classes in the population under investigation. Let now $\hat{f}_1$, $\hat{f}_2,\ldots,\hat{f}_J$ denote kernel estimators of a probability distribution density, calculated successively based on sets (3)-(5) treated as random samples (1) – a short description of the methodology used for their construction is contained in Section 2. In accordance with the classic Bayes approach [Duda et al, 2001], ensuring a minimum of expected value of losses, the classified element $\tilde{x}\in\mathbb{R}^n$ should then be given to the class for which the value

$$m_1\hat{f}_1(\tilde{x}),\ m_2\hat{f}_2(\tilde{x}),\ldots,m_J\hat{f}_J(\tilde{x}) \tag{6}$$

is the greatest. The above can be generalized by introducing to expressions (6) the positive coefficients z_1, $z_2,\ldots,z_J$:

$$z_1m_1\hat{f}_1(\tilde{x}),\ z_2m_2\hat{f}_2(\tilde{x}),\ldots,z_Jm_J\hat{f}_J(\tilde{x})\quad . \tag{7}$$

Taking as standard values $z_1=z_2=\ldots=z_J=1$, formula (7) brings us to (6). By appropriately increasing the value z_i, a decrease can be achieved in the probability of erroneously assigning elements of the *i*-th class to other wrong classes (although the danger does then exist of increasing the general number of misclassifications). Thanks to this, it is possible to favor classes which are in some way noticeable (e.g. in diagnostics, those representing faults causing large losses) or more heavily conditioned. For the classification task considered here, these are in natural way classes defined by nonstationary patterns – in the case of a significant difference in the speed of their changes, it is worth increasing coefficients relating to more varying patterns. The initial value 1.25 can be proposed for further research.

4 Discrete Derivative

The task of computing the value of the discrete derivative of the function $g:\mathbb{R}\to\mathbb{R}$ consists in calculating the quantity $g'(t)$ based on values of this function obtained for a finite number of the arguments $t_1,\ t_2,\ldots,t_k$. For the problem under investigation a backward derivative will be used, that is where $t=t_k$. As the task considered here does not require the differences between subsequent values $t_1,\ t_2,\ldots,t_k$ to be equal, it is therefore advantageous to apply interpolation methods. In the procedure worked out here, favorable results were achieved using a classic method based on Newton's interpolation polynomial. Detailed formulas are found in the survey article [Venter, 2010]. For the purposes of the procedure investigated in this paper, $k=3$ can be taken as a standard value.

5 Sensitivity Analysis for Learning Data

When modeling multidimensional problems using artificial neural networks, particular components of an input vector most often are characterized by diverse significance of information, and in consequence influence variously the result of the data processing. In order to eliminate superfluous – from the point of view of the investigated task – input vector components, a sensitivity analysis of the network with respect to particular learning data can be performed. As a result one obtains the parameters $\overline{S}_i$ describing proportionally the influence of the particular inputs ($i=1,2,\ldots,m$) on the output value, and then the least significant inputs can be eliminated.

Detailed description of the above procedure is found in the publications [Engelbrecht et al, 1995; Zurada, 1992].

6 Reducing Patterns' Size

In practice, some elements of sets (3)-(5), constituting patterns of particular classes, may have insignificant or even negative – in the sense of classification correctness – influence on quality of obtained results. Their elimination should therefore imply a reduction in the number of erroneous assignments, as well as decreasing calculation time. To this aim the sensitivity method for learning data, used in artificial neural networks, briefly noted in the previous section, will be applied.

To meet the requirements of this procedure, the definition of the kernel estimator will be generalized below with the introduction of the nonnegative coefficients w_1, $w_2,\ldots,w_m$, normed so that $\sum_{i=1}^{m} w_i = m$ and mapped to particular elements of random sample (1). The basic form of kernel estimator (2) then takes the form

$$\hat{f}(x) = \frac{1}{mh^n} \sum_{i=1}^{m} w_i K\left(\frac{x - x_i}{h} \right) \quad . \tag{8}$$

The coefficient w_i value may be interpreted as indicating the significance (weight) of the *i*-th element of the pattern to classification correctness.

The procedure for reducing patterns sets (3)-(5) consists – in its basic form – of two phases: of calculating the weight w_i, and then removing those elements of random sample (1), for which the respective weights have the lowest values. These tasks will subsequently be presented in the next two subsections.

6.1 Calculation of Weights w_i

In the method designed here, for the purpose of reduction of sets (3)-(5), separate neural networks are built for each investigated class.

The constructed network has three layers and is unidirectional, with m inputs (corresponding to particular elements of a pattern), a hidden layer whose size is equal to the integral part of the number $\sqrt{m}$, and also one output neuron. This network is submitted to a learning process using a data set comprising of the values of particular kernels for subsequent pattern elements, while the given output constitutes the value of the kernel estimator calculated for the pattern element under consideration. The network's learning is carried out using backward propagation of errors with momentum factor. On finishing this process, the thus obtained network undergoes sensitivity analysis on learning data, in accordance with the method presented in the previous section. The resulting coefficients $\overline{S}_i$ describing sensitivity, calculated in this way, constitute the fundament for calculating the preliminary values

$$\tilde{w}_i = \left(1 - \frac{\overline{S}_i}{\sum_{j=1}^{m} \overline{S}_j} \right) \quad , \tag{9}$$

after which they are normed to

$$w_i = m \frac{\tilde{w}_i}{\sum_{i=1}^{m} \tilde{w}_i} \quad . \tag{10}$$

The shape of formula (9) results from the fact that the network created here is the most sensitive to atypical and redundant elements, which – taking into account the form of kernel estimator (8) – implies a necessity to map the appropriately smaller values $\tilde{w}_i$, and in consequence w_i, to them. Coefficients (10) represent – as per the

idea presented while introducing the generalized form (8) – the significance of particular elements of the pattern to accuracy of the classification process.

6.2 Removal of Pattern Elements

Empirical research confirmed the natural assumption that the pattern set should be relieved of those elements for which $w_i < 1$. (Note that, thanks to normalization made by formula (10), the mean value of the coefficients w_i equals 1.)

7 Correcting the Smoothing Parameter and Modification Intensity Values

The classic universal methods of calculating the smoothing parameter value are often not proper for the classification task. This paper will propose a procedure suited to the conditioning of the investigated method of classification for nonstationary patterns, in particular those enabling successive adaptation with regard to the occurring changes.

Thus, it can be proposed to introduce $n+1$ multiplicative correcting coefficients for the values of the parameter defining the intensity of modification procedure c and smoothing parameters for particular coordinates h_1, $h_2, \ldots, h_n$, with respect to optimal ones calculated using the integrated square error criterion. Denote them as $b_0 \geq 0$, b_1, $b_2, \ldots, b_n > 0$, respectively. It is worth noticing that $b_0 = b_1 = \ldots = b_n = 1$ means in practice no correction. Next through a comprehensive search using a grid with a relatively large discretization value, one finds the most advantageous points regarding minimal incorrect classification sense. The final phase is a static optimization procedure in the $(n+1)$-dimensional space, where the initial conditions are the points chosen above, while the performance index is given as the number of misclassifications. This is an integer – to find the minimum a modified Hook-Jeeves algorithm [Kelley, 1999] was applied.

Following experimental research it was assumed that the grid used for primary searches has intersections at the points 0.25, $0.5, \ldots, 1.75$ for every coordinate. For such intersections the value of the performance index is calculated, after which the obtained results are sorted, and the 5 best become subsequent initial conditions for the Hook-Jeeves method, where the value of the initial step is taken as 0.2. After finishing every one of the above 5 "runs" of this method, the performance index value for the end point is calculated, and finally among them the one with the smallest value is shown.

Apart from the first step, the above procedure can be used in the simplified version, to successively specify current values for the correcting coefficients b_0, $b_1, \ldots, b_n$ as part of adaptation to changes in nonstationary conditions. To this end the Hook-Jeeves algorithm is used only once, taking the coefficients' previous values as initial conditions.

Finally it is worth noting that the correction of classification parameters is not necessary in this procedure. It does, however, increases classification accuracy and furthermore enables the use of a simplified method for calculation of smoothing parameters values, proposed in Section 2.

8 Classification Method for Nonstationary Patterns

This section, the most essential in this publication, presents the classification method for the nonstationary case, that is when all or some patterns of classes undergo significant – considering the investigated task – changes. Here, material presented in Sections 2-7 will be used. A block diagram of the calculation procedure is shown in Fig. 1. Blocks symbolizing operations performed on all elements of patterns (3)-(5) jointly are drawn with a continuous line, while a dashed line denotes operations on particular classes, and a dotted line – separate operations for each element of those patterns.

First one should fix the reference sizes of patterns (3)-(5), hereinafter denoted by m_1^*, $m_2^*, \ldots, m_J^*$. The patterns of these sizes will be the subject of a basic reduction procedure, described in Section 6. The sizes of patterns available at the beginning of the algorithm must not be smaller than the above referential values. These values can however be modified during the procedure's operation, with the natural condition that their potential growth does not increase the number of elements newly provided for the patterns. For preliminary research, $m_1^* = m_2^* = \ldots = m_J^* = 25 \cdot 2^n$ can be proposed. Lowering these values may worsen the classification quality, whereas an increase results in an excessive calculation time.

The elements of initial patterns (3)-(5) are provided as introductory data. Based on these – according to the procedures mentioned in Section 2 – the value of the parameter h is calculated (for the parameter c it is initially assumed to be equal 0.5). Figure 1 shows this action in block A. Next corrections in the parameters h and c values are made by taking the coefficients b_0, $b_1, \ldots, b_n$, as described in Section 7 (block B in Fig. 1).

The next procedure, shown by block C, is the calculation of the parameters w_i values mapped to particular patterns' elements, separately for each class, as in Section 6.1. Following this, within each class, the values of the parameter w_i are sorted (block D), and then – in block E – the appropriate m_1^*, $m_2^*, \ldots, m_J^*$ elements of the largest values w_i are designated to the classification phase itself. The remaining ones undergo further treatment, denoted in block U, which will be presented later, after Bayes classification has been dealt with.

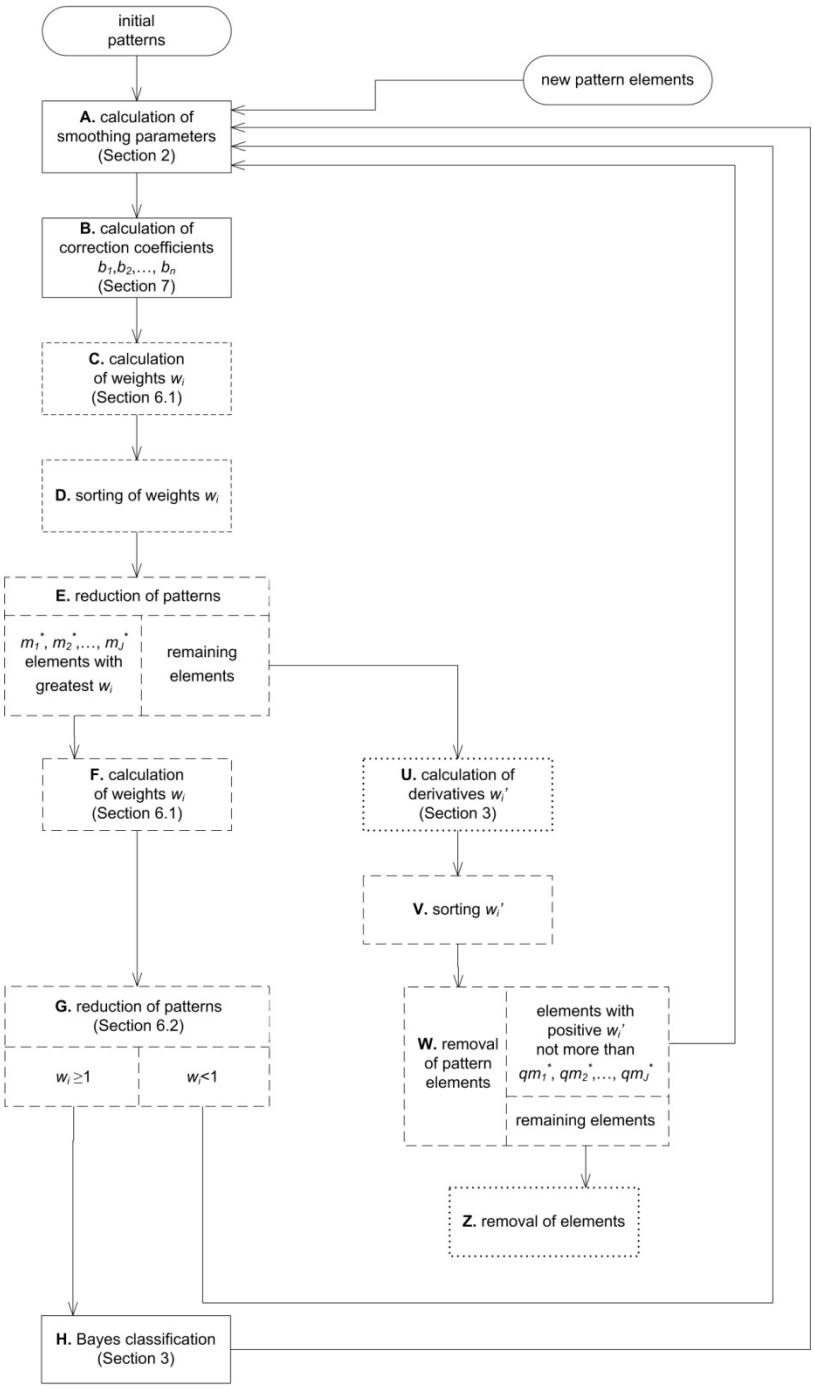

Fig. 1. Block diagram for classification algorithm

The reduced patterns separately go through a procedure newly calculating the values of parameters w_i, presented in Section 6.1 and depicted in block F. According to Section 6.2, as block G in Fig. 1 denotes, these patterns' elements for which $w_i \geq 1$ are submitted to further stages of the classification procedure, while those with $w_i < 1$ are sent to block A for further processing in the next steps of the algorithm, after adding new elements of patterns. The final, and also the principal part of the procedure worked out here is Bayes classification, presented in Section 3 and marked by block H. Obviously many tested elements $\tilde{x}$ can be subjected to classification separately. After the procedure has been finished, elements of patterns which have undergone classification are sent to the beginning of the algorithm to block A, to further avail of the next steps, following the addition of new elements of patterns.

Now – in reference to the end of the paragraph before the last – it remains to consider these patterns' elements, whose values w_i were not counted among the m_1^*, $m_2^*, \ldots, m_J^*$ largest for particular patterns. Thus, within block U, for each of them the derivative $w_i^{'}$ is calculated. If the element is "too new" and does not possess the $k-1$ previous values w_i, then the gaps are filled with zeros (because the values w_i generally oscillate around unity, such behavior significantly increases the derivative value, and in consequence ensures against premature elimination of this element). Next for each separate class, the elements $w_i^{'}$ are sorted (block V). As marked in block W, the respective

$$qm_1^*, \; qm_2^*, \ldots, qm_J^* \tag{11}$$

elements of each pattern with the largest derivative values, on the additional requirement that the value is positive, go back to block A for further calculations carried out after the addition of new elements. If the number of elements with positive derivative is less than $qm_1^*, \; qm_2^*, \ldots, qm_J^*$, then the number of elements going back may be smaller (including even zero). The remaining elements are permanently eliminated from the procedure, as shown in block Z. In the above notation q is a positive constant influencing the proportion of patterns' elements with little, but successively increasing meaning. The standard value of the parameter q can be proposed to be close to $k/2$, where k denotes the order of a discrete derivative, multiplied by the number of new elements added to the algorithm divided by the reference values, such however that $q \geq 0.05$; an increase in the parameter q value allows more effective conforming to pattern changes, although this potentially increases the calculation time, while lowering it may significantly worsen adaptation. In the general case this parameter can be different for particular patterns – then formula (11) takes the form

$$q_1 m_1^*, \; q_2 m_2^*, \ldots, q_J m_J^* \quad , \tag{12}$$

where $q_1, \; q_2, \ldots, q_J$ are positive.

The above procedure is repeated following the addition of new elements (block A in Fig. 1). Besides these elements – as has been mentioned earlier – for particular patterns respectively m_1^*, $m_2^*, \ldots, m_J^*$ elements of the greatest values w_i are taken, as well as up to qm_1^*, $qm_2^*, \ldots, qm_J^*$ (or in the generalized case $q_1 m_1^*$, $q_2 m_2^*, \ldots, q_J m_J^*$) elements of the greatest derivative w_i', so successively increasing its significance, most often due to the nonstationarity of patterns.

9 Empirical Verification and Comparison

The correct functioning and properties of the concept under investigation have been comprehensively verified numerically, and also compared with results obtained using a related procedure based on a support vector machine (SVM) method. Research was carried out for data sets in various configurations and with different properties, particularly with nonseparated classes, complex patterns, multimodal and consisting in detached subsets located alternately. Nonstationarity increased successively either in steps or periodically. The standard values of the parameters previously proposed in this paper were obtained through research carried out for verification purposes.

The following are the results obtained for a simple but representative case, enabling a telling illustration and interpretation of the procedure summaries in Section 8. For visual purposes the two dimensional space ($n = 2$) and the two classes ($J = 2$) will be used. For both classes, the patterns begin with 100 elements ($m_1 = m_2 = 100$), obtained using a generator with normal distribution with the unique variance. The expected value of the first – stationary – class is located permanently in the origin of the space $\mathbb{R}^2$, while for the second – nonstationary – following an initial period of no movement, encircles it with the radius 3, adding 10 new elements every 10 degrees before coming to a stop in its original location. According to the suggestions formulated earlier, it was also assumed $m_1^* = m_2^* = 100$ and $q = 0.2$.

Figure 2 illustrates the number of misclassifications in a typical course of a procedure created in this paper. From the beginning, up to step 18, the second class is invariable. First a slight increase in the number of erroneous classifications occurs – in every step around 10% new elements of patterns are added, which worsens the working conditions for the neural network. Finally, however, once the patterns are stabilized, the number of misclassifications settles at the level 0.08. In step 18 the aforementioned orbital movement of the second class begins. First the number of erroneous classifications rises to around 0.12, and then – after the kernels, which were not previously removed due solely to a positive derivative w_i', have received the appropriate meaning – the number of misclassifications drops and levels off at 0.105. In step 54, where the second class stops, occurrences similar to the above take place, when the number of classification errors returns to its initial level of 0.08.

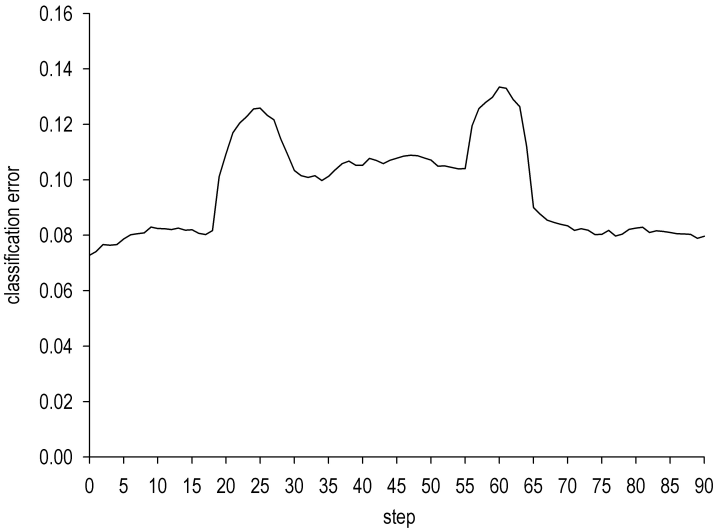

Fig. 2. Typical course using the investigated procedure ($z_1 = 1$, $z_2 = 1$)

Thanks to the generalization of formula (6) to (7), the classification quality can increase by mapping to the classes with nonstationary patterns greater values of the coefficients z_i . Figure 3 shows the results obtained with $z_1 = 1$ and $z_2 = 1.25$. It is worth noting that the total number of misclassifications lowered with respect to that obtained for the basic case in Fig. 2. This especially concerns local maximums existing after the second class starts and stops moving (steps 18 and 54).

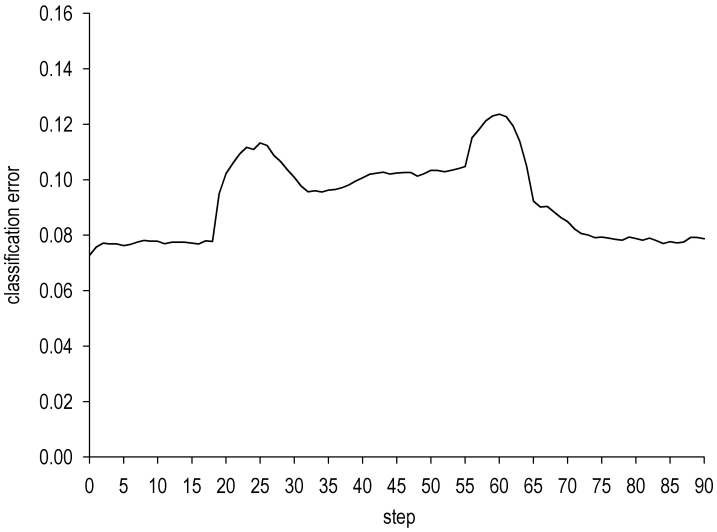

Fig. 3. Course with differing values of the coefficients z_i : $z_1 = 1$, $z_2 = 1.25$

The procedure worked out and described here was compared with a method based on the support vector machine (SVM) concept, presented in the publication [Krasotkina et al, 2011], taken as the closest regarding the conditioning considered in this paper research task. The obtained results are shown in Fig. 4 – they were achieved in conditions identical to Fig. 3, with which they will be compared. Although in conditions of stationarity of the second pattern (steps before 18 and after 54) the number of misclassifications leveled off at 0.08, in the case using the SVM, however, starts at 0.10, instead of 0.07 as in the procedure investigated here (compare Fig. 3 and 4). When the second pattern changes (steps 18-54) the amount of errors generated by the SVM settles at the level 0.12, or even slightly higher than that of local maximums appearing in the method presented in this paper after steps 18 and 54 (compare again Fig. 3 and 4). It should be underlined, though, that when the second class is moving, the number of misclassifications does not fall to the level $0.1-0.105$ (Fig. 4), as is the case with the procedure worked out here (Fig. 3). Thus one can see that the concept method in this paper has an advantage over the SVM procedure, especially in conditions of gradual change. Taking into account the fact that its idea is based on derivatives of a predictive nature, this observation is completely understandable.

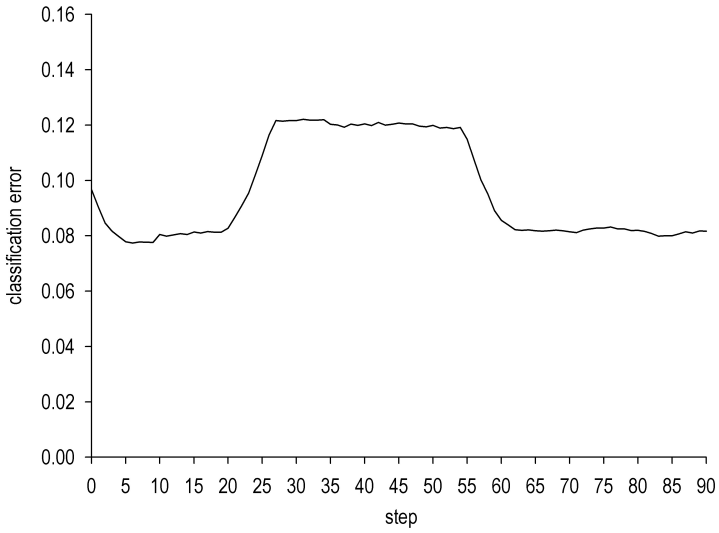

Fig. 4. Course using the SVM method

To summarize: numerical testing wholly confirmed the positive features of the method worked out. In particular, the results show that the classifying algorithm can be used successfully for inseparable classes of complex multimodal patterns as well as for those consisting of incoherent subsets at alternate locations. The examined nonstationarity increased successively, and was periodical as well as occurring in steps. For the former type, the procedure presented in this paper proved to be particularly advantageous and useful.

10 Final Comments and Remarks

This paper presented a classification procedure which allows for nonstationarity of patterns and successive supply of new elements to them. Neither the number of classes itself, nor the number of nonstationary ones are methodologically limited. The concept is based on the Bayes approach, which allows for the minimization of expected loss value arising from erroneous classifications, as well as actively influencing the proportion of probabilities of classification errors between particular classes. The use of kernel estimators frees the algorithm from patterns' shapes. The procedure operation is based on the sensitivity method used with artificial neural networks. It enables the removal of those elements of patterns, which are of insignificant or even negative influence on accuracy of results. However, it retains for further calculation some of these elements which due to nonstationarity successively increase their positive impact. Appropriate adaptation is also performed on classifier parameters.

A broad description of the presented method – in particular the full set of formulas – is contained in the paper [Kulczycki and Kowalski, 2013b] which will appear soon.

References

1. Duda, R.O., Hart, P.E., Storck, D.G.: Pattern Classification. Wiley, New York (2001)
2. Engelbrecht, A.P., Cloete, I., Zurada, J.: Determining the Significance of Input Parameters Using Sensitivity Analysis. In: Sandoval, F., Mira, J. (eds.) IWANN 1995. LNCS, vol. 930, pp. 382–388. Springer, Heidelberg (1995)
3. Han, J., Kamber, M.: Data Mining: Concepts and Techniques. Wiley, New York (2001)
4. Kelley, C.T.: Iterative Methods for Optimization. SIAM, Philadelphia (1999)
5. Krasotkina, O.V., Mottl, V.V., Turkov, P.A.: Bayesian Approach to the Pattern Recognition Problem in Nonstationary Environment. In: Kuznetsov, S.O., Mandal, D.P., Kundu, M.K., Pal, S.K. (eds.) PReMI 2011. LNCS, vol. 6744, pp. 24–29. Springer, Heidelberg (2011)
6. Kulczycki, P.: Estymatory jadrowe w analizie systemowej. WNT, Warsaw (2005)
7. Kulczycki, P., Hryniewicz, O., Kacprzyk, J. (eds.): Techniki informacyjne w badaniach systemowych. WNT, Warsaw (2007)
8. Kulczycki, P., Kowalski, P.A.: Bayes classification of imprecise information of interval type. Control and Cybernetics 40, 101–123 (2011)
9. Kulczycki, P., Kowalski, P.A.: Klasyfikacja Bayesowska przy Niestacjonarnych Wzorcach. In: Proceedings of the 11th International Conference on Diagnostics of Processes and Systems, Lagow Lubuski, Poland, September 8-11, paper_4 (2013a)
10. Kulczycki, P., Kowalski, P.A.: Bayes Classification for Non-Stationary Patterns (in press, 2013b)
11. Silverman, B.W.: Density Estimation for Statistics and Data Analysis. Chapman and Hall, London (1986)
12. Venter, G.: Review of Optimization Techniques. In: Encyclopedia of Aerospace Engineering, pp. 5229–5238. Wiley, New York (2010)
13. Wand, M.P., Jones, M.C.: Kernel Smoothing. Chapman and Hall, London (1995)
14. Zurada, J.: Introduction to Artificial Neural Neural Systems. West Publishing, St. Paul (1992)