JAISCR, 2020, Vol. 10, No. 1, pp. 47 – 56
10.2478/jaiscr-2020-0004

ROUGH SUPPORT VECTOR MACHINE FOR CLASSIFICATION WITH INTERVAL AND INCOMPLETE DATA

Robert K. Nowicki[1,*], Konrad Grzanek [2], Yoichi Hayashi[3]

[1]*Department of Computer Engineering, Częstochowa University of Technology,*
42–200 Częstochowa, Poland

[2]*Information Technology Institute, University of Social Science,*
Łódź, Poland
Clark University, Worcester, MA 01610, USA

[3]*Department of Computer Science, Meiji University, Kawasaki 214-8571 Japan*

[*]*E-mail: robert.nowicki@pcz.pl*

Submitted: 23th July 2019; Accepted: 19th November 2019

Abstract

The paper presents the idea of connecting the concepts of the Vapnik's support vector machine with Pawlak's rough sets in one classification scheme. The hybrid system will be applied to classifying data in the form of intervals and with missing values [1]. Both situations will be treated as a cause of dividing input space into equivalence classes. Then, the SVM procedure will lead to a classification of input data into rough sets of the desired classes, i.e. to their positive, boundary or negative regions. Such a form of answer is also called a three–way decision. The proposed solution will be tested using several popular benchmarks.

Keywords: support vector machines, rough sets, missing features, interval data, three–way decision

1 Introduction

The support vector machine (SVM) algorithm has been proposed by Vladimir Vapnik [15]. It is, in general, a linear classifier which separates objects $x(\tau)$, where $\tau = 1, \ldots, \tau_{\max}$, described by vector $\mathbf{v}(\tau) \in V \subset \mathfrak{R}^n$ into classes by $n-1$ dimensional hyperplanes defined by selected reference samples x^r_{ref}. The samples are called support vectors. The method is a direct continuation of the first pattern recognition algorithm proposed by Fisher [3]. It can also be perceived in reference to k nearest neighbor method. The SVM allows for limiting the number of reference samples stored and processed during a classification process to a small number of support vectors. Selection of the vectors is realized through the maximization of the margin between the hyperplane described by a decision function and the reference samples. In the case of classes which are not linearly separable, an additional kernel function K is applied. Depending on the interpretation, it transforms input space V or transforms the hyperplanes into other separating shapes [14, 5, 13].

The rough set theory has been proposed by Zdzisław Pawlak [11, 12]. The theory defines the space of approximation in which groups of objects are indistinguishable within a given level of description. The groups are called equivalence classes or atoms. The level of description and, as a consequence, a division of the input space into atoms is an effect of limited information or imperfection of information about particular objects. Thus, the classifier classifies not particular objects but whole atoms. It builds classes from atoms as it is done with bricks. Particular atoms can belong to the class as a whole; they can be entirely outside the class or partially belong to it. Similarly, there are three possible results of classification. The object and its atom can belong to the class, not belong there, or the result is indefinite.

The general idea of combining the classic classification systems with the rough set theory has been presented in [10]. It contains a description of a few rough set based hybrid systems. There are rough fuzzy and fuzzy rough classification systems, rough neural network classifiers, and the rough nearest neighbor classifier. They have been tested in iterative and ensemble modes.

In the existing research, the concept of rough support vector machine which works in such way has not been considered, yet. Motivated by the above fact, in this paper the combination of both ideas is considered. The proposed rough support vector machine, which is the subject of the paper, is also the next step in developing rough set based hybrid systems.

The rough support vector machine classifier proposed in the paper has some properties common for all rough set based hybrid systems mentioned above. The object under classification is described by values or intervals. The width of intervals represents imperfection of the description. The interval which covers the whole domain of an attribute represents the missing value. It is assumed that the intervals always contain the true value of an attribute. In this class of the classifying systems, a misclassification is never the result of the imperfection of the object description. The change of width of intervals never results in a change of classification. However, it can lead to an indefinite result. This property positively distinguished the rough set–based hybrid systems from all solutions based on marginalization and imputation.

There are many modifications of the support vector machines algorithm. They allow realizing the non–linear and multi–class classification [4, 18] or improve the algorithm performance [6, 7, 19]. The rough set based hybrid system proposed in this paper is based on the basic version of the algorithm. However, there are no obstacles to transfer the idea to other versions of the algorithm.

The paper is organized as follows. Section 2 contains a brief description of the basic version of the support vector machines. Section 3 covers the main idea presented in the paper, i.e. the rough support vector machine. It begins with some formal elements coming from rough set theory, and then the hybrid system is defined. The performance of classification is illustrated in Section 4 with the use of a few classic benchmarks. The final Section contains conclusions and remarks. Plans for further research on the subject are outlined as well.

2 Classic support vector machines

In the basic version of the method, two linearly separable classes are considered. More generally, the separation of a single class among all others is taken into account — the ”one against all” problem. The aim is to determine the hyperplane

$$\mathbf{w}^T\mathbf{v} + w_0 = 0, \tag{1}$$

which realises the separation [2, 9, 15, 16]. In the multiclass classification, special versions of the algorithm have been developed [17]. However, the individual hyperplane determined for each class can be also applied.

The parameters of hyperplane (1) should be adjusted to separate the samples from class ω_j from the other ones and to maximize the distance between the hyperplane and the closest samples from both classes. Moreover, the distance should be equal for both classes. To satisfy such requirements two hyperplanes parallel to (1) are defined as follows

$$\mathbf{w}^T\mathbf{v} + w_0 - c = 0 \tag{2}$$

and

$$\mathbf{w}^T\mathbf{v} + w_0 + c = 0. \tag{3}$$

For particular objects $x(\tau)$, the following conditions must be met

$$\mathbf{w}^T\mathbf{v}(\tau) + w_0 - c \geq 0 \text{ for } x(\tau) \in \omega_j \tag{4}$$

and

$$\mathbf{w}^T\mathbf{v}(\tau) + w_0 + c \leq 0 \text{ for } x(\tau) \notin \omega_j. \tag{5}$$

Condition (4) requests that all samples that belong to class ω_j are on one side of hyperplane (2) or lie on it. Condition (5) requests that all samples that do not belong to class ω_j are on one side of hyperplane (3) or lie on it. Because hyperplane (1) is the separation boundary, both condition must be met also for $c = 0$. The samples which lay on hyperplanes (2) and (3) are the support vectors.

The distance between hyperplanes (2) and (3) (the margin of separation) is equal to

$$\rho = \frac{2c}{\|\mathbf{w}\|}. \tag{6}$$

Thus, the goal is to find parameters $\mathbf{w}$, w_0, and c which meet conditions (4) and (5) and maximize distance (6).

$$\begin{gathered} Q_j\left(\mathbf{w}, w_0\right) = \tfrac{1}{2}\mathbf{w}^T\mathbf{w} \\ -\textstyle\sum_{t=1}^{t_{\max}} \alpha_t\left(d_j(\tau)\left(\mathbf{w}^T\mathbf{v}(\tau) + w_0\right) - c\right), \end{gathered} \tag{7}$$

where $d_j(\tau) = 1$ for $x(\tau) \in \omega_j$ and $d_j(\tau) = -1$ for $x(\tau) \notin \omega_j$. The optimization algorithm looks for parameters $\mathbf{w}$ and w_0 which minimize Q_j and Lagrange multipliers α_t maximizing the function. The non–zero multipliers determine the support vectors. All others can be omitted in the SVM classifier.

When we assume that $c = 1$, the task of minimizing Q_j can be solved in two steps. Firstly, the algorithm should find non–negative values of multipliers α_t which maximize the following expression

$$Q_{\alpha j} = -\sum_{t=1}^{t_{\max}}\sum_{t'=1}^{t_{\max}} \alpha_t\alpha_{t'}d_j(\tau)d_j(\tau')\mathbf{v}_{t'}^T\mathbf{v}_t + \sum_{t=1}^{t_{\max}}\alpha_t \tag{8}$$

and fulfill the restriction

$$\sum_{t=0}^{m} \alpha_t d^j(\tau) = 0. \tag{9}$$

Secondly, coefficients $\mathbf{w}$ and w_0 are determined

$$\mathbf{w} = \sum_{t=1}^{t_{\max}} \alpha_t d(\tau)\mathbf{v}(\tau). \tag{10}$$

The decision of the SVM classifier is expressed as follows

$$\begin{gathered} z_j(\tau) = sgn\left(\mathbf{w}^T\mathbf{v}(\tau) + w_0\right) = \\ = sgn\left(\textstyle\sum_{r:\, x_{\text{ref}}^r \in SV} \alpha_t d_j^r(\tau)\mathbf{v}_{\text{ref}}^{rT}\mathbf{v}(\tau) + w_0\right). \end{gathered} \tag{11}$$

When the classes are not linearly separable, the transformation by a kernel function is applied and the decision is expressed by

$$z_j(\tau) = sgn\left(\sum_{r:\, x_{\text{ref}}^r \in SV} \alpha^r d_j^r K\left(\mathbf{v}_{\text{ref}}^{rT}\mathbf{v}(\tau)\right) + w_0\right). \tag{12}$$

3 Rough support vector machines

The general form of the SVM classifier presented above works properly when the description of object $x(\tau)$ under classification described by a vector of features values $\mathbf{v}(\tau)$ as well as elements of support vectors $\mathbf{v}_{\text{ref}}$ are real values — $v_j(\tau) \in \Re \cap \left[v_{i\min}, v_{i\max}\right]$ and $v_{\text{ref}i}^r \in \Re \cap \left[v_{i\min}, v_{i\max}\right]$. In such case the decision of the classifier is unequivocal. However, when in vector $\mathbf{v}(\tau)$ or vector $\mathbf{v}_{\text{ref}}$ at least one interval or unknown value occurs, the decision might be not unequivocal because the true value of such feature is unknown and can take any value inside interval $\overline{\underline{v}}_i(\tau) = \left[\underline{v_i}(\tau), \overline{v_i}(\tau)\right]$ or even $\left[v_{i\min}, v_{i\max}\right]$. The decision is unequivocal only when for all values in the interval the classifications are the same. However, in general, it is not possible to test an infinite number of values. The set of features with known and real values is denoted by P, the set of features with unknown or interval values — by G. The set of all considered features is denoted by $Q = P \cup G$. V_i is the set of the values obtained by feature v_i, so $V_i = \left[v_{i\min}, v_{i\max}\right]$.

The goal of the proposed solution is to adapt the SVM classifier to work with interval data. In such case the object is classified into rough classes $\left\{\underline{\tilde{P}}\omega_j, \overline{\tilde{P}}\omega_j\right\}$, which are approximations of classes ω_j. As a result, the object is classified to one of the three regions of class ω_j:

- positive region $Pos\left(\omega_j\right)$,
- negative region $Neg\left(\omega_j\right)$,
- boundary region $Bnd\left(\omega_j\right)$.

Thus, the rough SVM classifier should find the lowest and the highest values of $z_j(\tau)$, expressed in equation (11), which can be obtained for all values $v_i(\tau)$ from interval $\overline{\underline{v}}_i(\tau) = \left[\underline{v_i}(\tau), \overline{v_i}(\tau)\right]$. The relationship between $z_j(\tau)$ and $v_i(\tau)$ is monotonous and the derivative of expression inside function *sgn* with respect to $v_i(\tau)$ is constant, i.e.

$$\frac{\partial \mathbf{w}^T \mathbf{v}(\tau) + w_0}{\partial v_i(\tau)} = w_i, \tag{13}$$

where $w_i = \sum_{r:\, x_{\text{ref}}^r \in SV} \alpha^r d_j^r v_{\text{ref}i}^r$.

Thanks to this, the answer of the rough SVM can be expressed by interval $[\underline{z}(\tau), \overline{z}(\tau)]$ calculated using only the lowest and the highest values in intervals $\overline{\underline{v}}_i(\tau)$. Thus, the left bound of the answer is calculated by the expression

$$\begin{aligned} \underline{z}(\tau) &= sgn\left(\mathbf{w}^T \mathbf{v}_*(\tau) + w_0\right) = \\ &= sgn\left(\textstyle\sum_{r:\, x_{\text{ref}}^r} \alpha^r d^r \mathbf{v}_{\text{ref}}^{rT} \mathbf{v}_*(\tau) + w_0\right) \end{aligned} \tag{14}$$

and the right bound as follows

$$\begin{aligned} \overline{z}(\tau) &= sgn\left(\mathbf{w}^T \mathbf{v}^*(\tau) + w_0\right) = \\ &= sgn\left(\textstyle\sum_{r:\, x_{\text{ref}}^r} \alpha^r d^r \mathbf{v}_{\text{ref}}^{rT} \mathbf{v}^*(\tau) + w_0\right), \end{aligned} \tag{15}$$

where $\mathbf{v}_*(\tau) = [v_{1*}(\tau), \ldots, v_{n*}(\tau)]$, $\mathbf{v}^*(\tau) = [v_1^*(\tau), \ldots, v_n^*(\tau)]$, and

$$v_{*i}(\tau) = \begin{cases} \underline{v}_i(\tau) & \text{if } w_i > 0 \\ \overline{v}_i(\tau) & \text{if } w_i < 0 \end{cases} \tag{16}$$

$$v_i^*(\tau) = \begin{cases} \underline{v}_i(\tau) & \text{if } w_i < 0 \\ \overline{v}_i(\tau) & \text{if } w_i > 0 \end{cases} \tag{17}$$

In the case of using kernel K to realize non–linear classification, the answer in the form of interval $[\underline{z}(\tau), \overline{z}(\tau)]$ is expressed in a similar way, i.e.

$$\underline{z}(\tau) = sgn\left(\sum_{r:\, x_{\text{ref}}^r} \alpha^r d^r K\left(\mathbf{v}_{\text{ref}}^r, \mathbf{v}_*(\tau)\right) + w_0\right) \tag{18}$$

and

$$\overline{z}(\tau) = sgn\left(\sum_{r:\, x_{\text{ref}}^r} \alpha^r d^r K\left(\mathbf{v}_{\text{ref}}^r, \mathbf{v}^*(\tau)\right) + w_0\right). \tag{19}$$

However, the selection of values $\mathbf{v}_*(\tau)$ and $\mathbf{v}^*(\tau)$ is more complicated and is not restricted to two boundary values $\underline{v}_i(\tau)$ and $\overline{v}_i(\tau)$. They should meet the conditions

$$K\left(\mathbf{v}_{\text{ref}}^{rT}, \mathbf{v}_*(\tau)\right) = \inf_{\substack{\hat{v}_i \in \left[\underline{v_i}(\tau), \overline{v_i}(\tau)\right] \\ i:\, c_i \in G}} K\left(\mathbf{v}_{\text{ref}}^{rT}, \hat{\mathbf{v}}(\tau)\right), \tag{20}$$

$$K\left(\mathbf{v}_{\text{ref}}^{rT}, \mathbf{v}^*(\tau)\right) = \sup_{\substack{\hat{v}_i \in \left[\underline{v_i}(\tau), \overline{v_i}(\tau)\right] \\ i:\, c_i \in G}} K\left(\mathbf{v}_{\text{ref}}^{rT}, \hat{\mathbf{v}}(\tau)\right), \tag{21}$$

where $\hat{\mathbf{v}}(\tau) = [\hat{v}_1, \ldots, \hat{v}_n]$, but for $c_i \in P$ value $\hat{v}_i$ is just equal to $v_i(\tau)$.

In the case of the Gaussian kernel

$$K\left(\mathbf{v}_{\text{ref}}^r, \mathbf{v}(\tau)\right) = e^{-\beta\left|\mathbf{v}_{\text{ref}}^r - \mathbf{v}(\tau)\right|^p} \tag{22}$$

value v_{*i} is selected from two–element set $\left\{\underline{v_i}(\tau), \overline{v_i}(\tau)\right\}$, which can be written as follows

$$K\left(\mathbf{v}_{\text{ref}}^{rT}, \mathbf{v}(\tau)\right) = \inf_{\substack{\hat{v}_i \in \left\{\underline{v_i}(\tau), \overline{v_i}(\tau)\right\} \\ i:\, c_i \in G}} K\left(\mathbf{v}_{\text{ref}}^{rT}, \hat{\mathbf{v}}(\tau)\right), \tag{23}$$

and value v_i^* obtains value $v_{\text{ref}i}^r$ if only $v_{\text{ref}i^r} \in \left[\underline{v_i}(\tau), \overline{v_i}(\tau)\right]$, thus

$$K\left(\mathbf{v}_{\text{ref}}^{rT}, \mathbf{v}(\tau)\right) = K\left(\mathbf{v}_{\text{ref}}^{rT}, \hat{\mathbf{v}}(\tau)\right) : \hat{v}_i = v_{\text{ref}i}^r, i\colon c_i \in G. \tag{24}$$

As a result of applying function *sgn*, in equations (14), (15), (18), and (19), values $\underline{z}_j(\tau)$ and $\overline{z}_j(\tau)$ are from set $\{-1, 0, 1\}$. Obviously, inequality $\underline{z}_j(\tau) \leq \overline{z}_j(\tau)$ occurs. Finally, depending on interval $\left[\underline{z}_j(\tau), \overline{z}_j(\tau)\right]$, object $x(\tau)$ is classified to one of the regions,m i.e. the positive, negative or boundary region of class ω_j, as follows

$$x(\tau) \in \begin{cases} Pos_P(\omega_j) & \text{if } \left[\underline{z}_j, \overline{z}_j\right] = [1, 1] \\ Neg_P(\omega_j) & \text{if } \left[\underline{z}_j, \overline{z}_j\right] = [-1, -1] \\ Bnd_P(\omega_j) & \text{if } 0 \in \left[\underline{z}_j, \overline{z}_j\right] \end{cases}. \tag{25}$$

The complete missing values can be treated in the same way as the values in the form of the interval. In such a solution the knowledge about the range of variation of subsequent features is applied, i.e.

$$c_t \in P \Rightarrow \left[\underline{v}_i(\tau), \overline{v}_i(\tau)\right] = \left[v_{i\min}, v_{i\max}\right]. \tag{26}$$

It is worth noting that the behavior of the classifier in the case of missing values should take into account the nature of such lack. Generally, the three types of missing values are distinguished [8] depending on the additional knowledge about the reason and relationships between the features with missing values and others, and they are MCAR (Missing Completely At Random), MAR (Missing At Random), and MNAR (Missing Not At Random). This knowledge can be used to narrowing intervals $[\underline{v}_i(\tau), \overline{v}_i(\tau)]$. Both missing and interval values can occur and be served by the rough SVM classifier simultaneously.

4 Illustrative examples

The proposed rough support vector machine classifier has been tested using classic benchmarks taken from the machine learning repository of the University of California at Irvine. The three of them have been selected for presentation, i.e. Glass Identification, Pima Indian Diabetes diagnostics and Wisconsin Breast Cancer diagnostics. They have 9, 9, and 8 input attributes, respectively. The two–class versions have been used. The support vector machines have been set for each benchmark when all the values of their attributes were represented by real numbers. During classification, the input values are presented in the form of intervals of various widths, which were generated around the true values. The missing values are simulated for all combinations of input attributes. All the experiments were performed in the 10–fold cross–validation mode.

Tables 1–3 contain the results for the experiments with various width of input intervals for subsequent benchmarks including both reference and test samples. The data in the tables as well as the visualization presented in Figures 1, 3 and 5 show that in the case of reference samples the classifiers are never wrong, but refuse to respond when the width of interval values is too high. However, the acceptable width is different for particular samples. The cases of misclassification occur for test samples. When the width of input values is high, particular samples remain without a decision. It applies to samples which are previously classified correctly as well as incorrectly.

Table 1. The results of the Glass Identification test using the rough support vector classifier

Width of intervals [%]	Decisions [%]		
	correct	incorrect	no decision
0	100.0/92.1	0.0/7.9	0.0/0.0
1	100.0/91.1	0.0/7.5	0.0/1.4
2	100.0/90.2	0.0/6.5	0.0/3.3
4	88.3/85.0	0.0/5.6	11.7/9.3
8	64.8/65.0	0.0/3.7	35.2/31.3
16	17.9/16.4	0.0/0.9	82.1/82.7
32	4.0/5.1	0.0/0.0	96.0/94.9
64	0.0/0.0	0.0/0.0	100.0/100.0

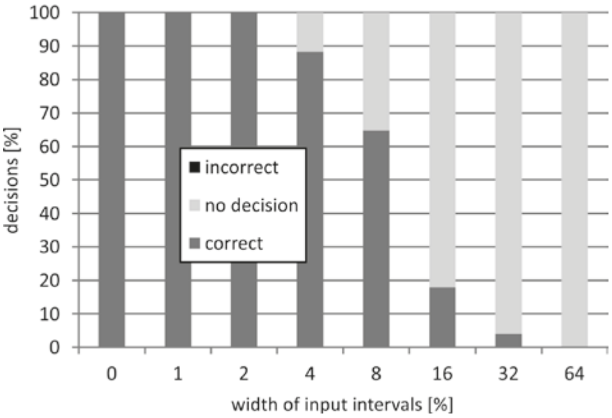

Figure 1. The results of the Glass Identification test using the rough support vector classifier — reference samples

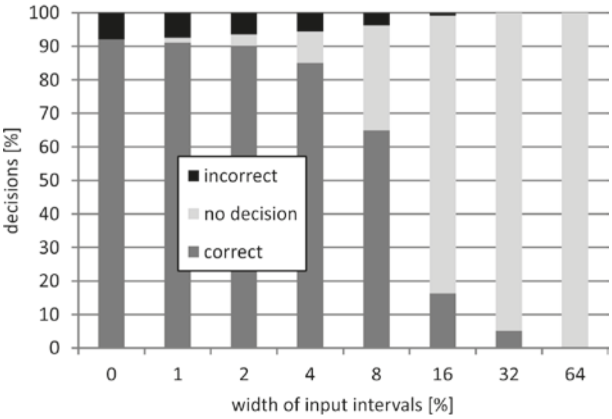

Figure 2. The results of the Glass Identification test using the rough support vector classifier — testing samples

Table 2. The results of Pima Indian Diabetes diagnostics using the rough support vector classifier

Width of intervals [%]	Decisions [%]		
	correct	incorrect	no decision
0	100.0/65.6	0.0/34.4	0.0/0.0
1	75.4/59.0	0.0/27.3	24.6/13.7
2	55.3/52.5	0.0/20.6	44.7/27.0
4	38.2/37.5	0.0/14.1	61.8/48.4
8	15.2/14.5	0.0/4.8	84.8/80.7
16	1.3/1.2	0.0/0.7	98.7/98.2
32	0.0/0.1	0.0/0.0	100.0/99.9
64	0.0/0.0	0.0/0.0	100.0/100.0

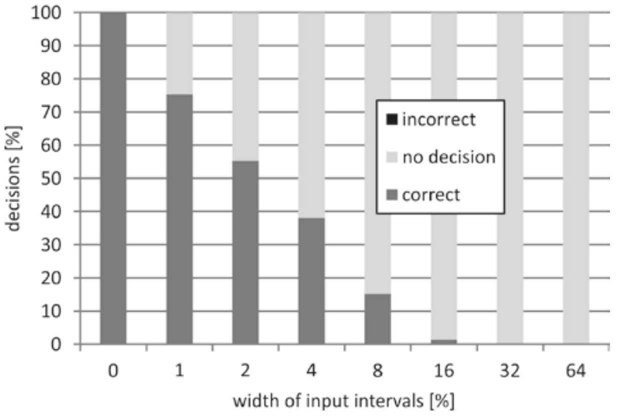

Figure 3. The results of Pima Indian Diabetes diagnostics using the rough support vector classifier — reference samples

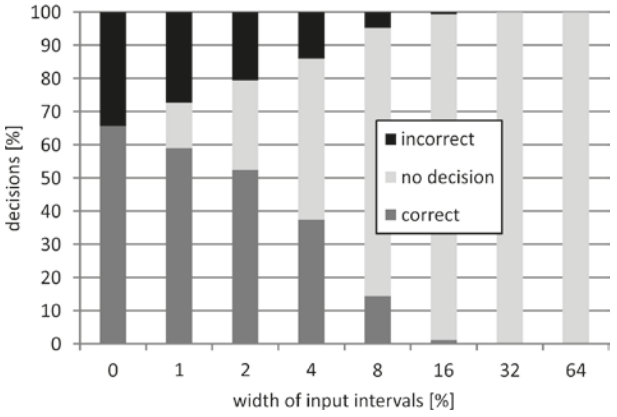

Figure 4. The results of Pima Indian Diabetes diagnostics using the rough support vector classifier — testing samples

Table 3. The results of Wisconsin Breast Cancer diagnostics using the rough support vector classifier

Width of intervals [%]	Decisions [%]		
	correct	incorrect	no decision
0	100.0/95.0	0.0/5.0	0.0/0.0
1	100.0/94.6	0.0/4.7	0.0/0.7
2	100.0/94.1	0.0/4.5	0.0/1.3
4	100.0/92.8	0.0/3.7	0.0/3.5
8	94.1/90.2	0.0/1.6	5.9/8.2
16	81.4/80.8	0.0/0.4	18.6/18.7
32	16.5/15.4	0.0/0.0	83.5/84.6
64	4.9/4.8	0.0/0.0	95.1/95.2

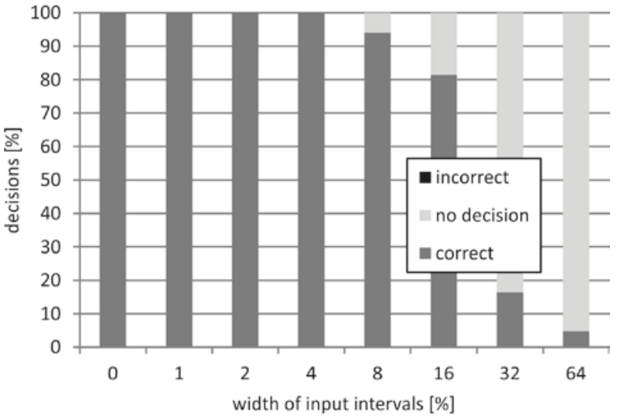

Figure 5. The results of Wisconsin Breast Cancer diagnostics using the rough support vector classifier — reference samples

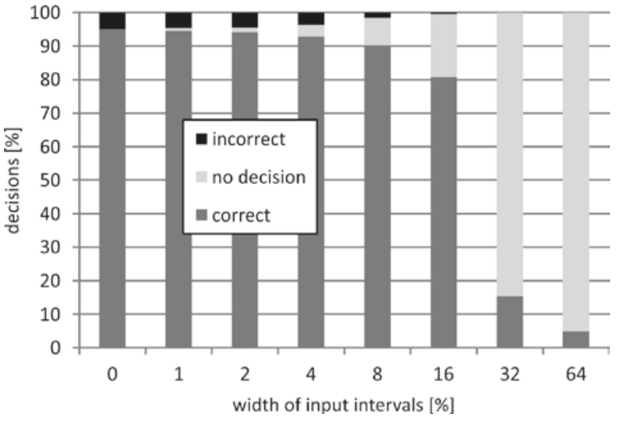

Figure 6. The results of Wisconsin Breast Cancer diagnostics using the rough support vector classifier — testing samples

Tables 4–6 contain the results for the tests with missing values. They have been visualized in Figures 7–11. The experiments have been performed with the same datasets and the same classifiers as the previous ons. So, the classifiers are never wrong in the case of reference samples as in the previous tests. The results show that even single missing values leads to an indefinite answer. This distinguished negatively the rough set support vector machine against other rough set–based hybrid systems described in [10].

Table 4. The results of the Glass Identification test using the rough support vector classifier

Number of missing values	Decisions [%]		
	correct	incorrect	no decision
0	100.0/92.1	0.0/7.9	0.0/0.0
1	6.4/6.5	0.0/0.0	93.6/93.5
2-9	0.0/0.0	0.0/0.0	100.0/100.0

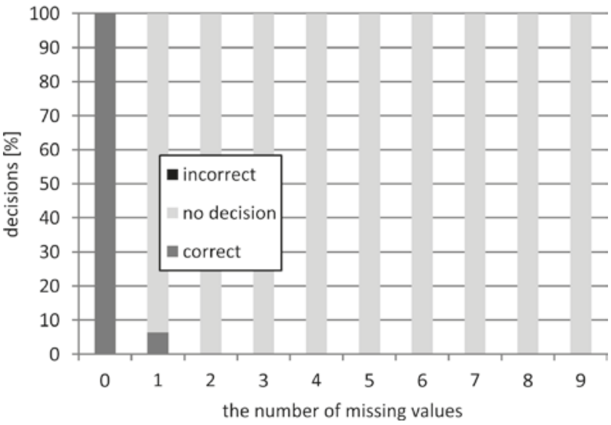

Figure 7. The results of the Glass Identification test using the rough support vector classifier — reference samples

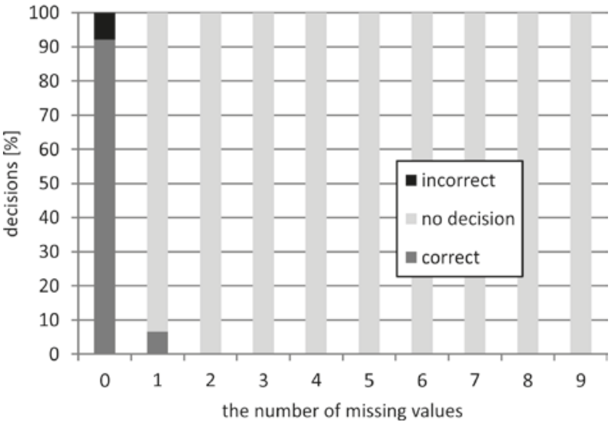

Figure 8. The results of the Glass Identification test using the rough support vector classifier — testing samples

Table 5. The results of Pima Indian Diabetes diagnostics using the rough support vector classifier

Number of missing values	Decisions [%]		
	correct	incorrect	no decision
0	100.0/95.0	0.0/5.0	0.0/0.0
1	21.6/21.7	0.0/0.0	78.4/78.3
2	6.4/6.6	0.0/0.0	93.6/93.4
3	1.6/1.6	0.0/0.0	98.4/98.4
4	0.2/0.3	0.0/0.0	99.8/99.7
5-9	0.0/0.0	0.0/0.0	100.0/100.0

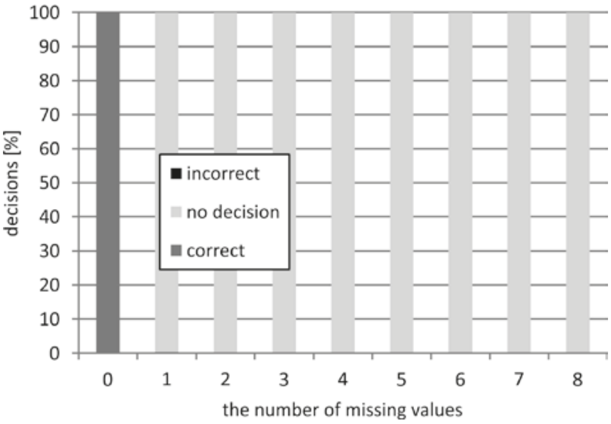

Figure 9. The results of Pima Indian Diabetes diagnostics using the rough support vector classifier — reference samples

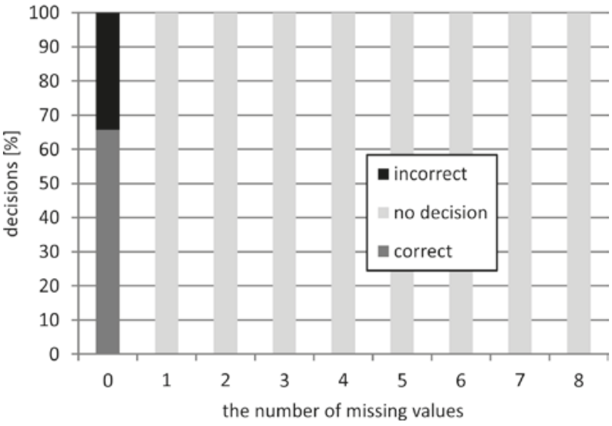

Figure 10. The results of Pima Indian Diabetes diagnostics using the rough support vector classifier — testing samples

Table 6. The results of Wisconsin Breast Cancer diagnostics using the rough support vector classifier

Number of missing values	Decisions [%]		
	correct	incorrect	no decision
0	100.0/65.6	0.0/34.4	0.0/0.0
1	0.3/0.3	0.0/0.0	99.7/99.7
2-8	0.0/0.0	0.0/0.0	100.0/100.0

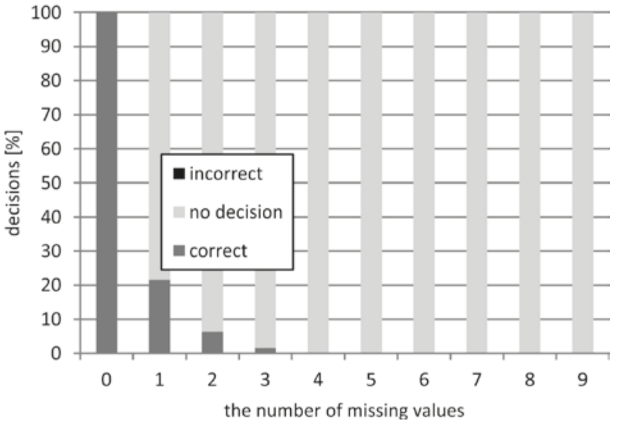

Figure 11. The results of Wisconsin Breast Cancer diagnostics using the rough support vector classifier — reference samples

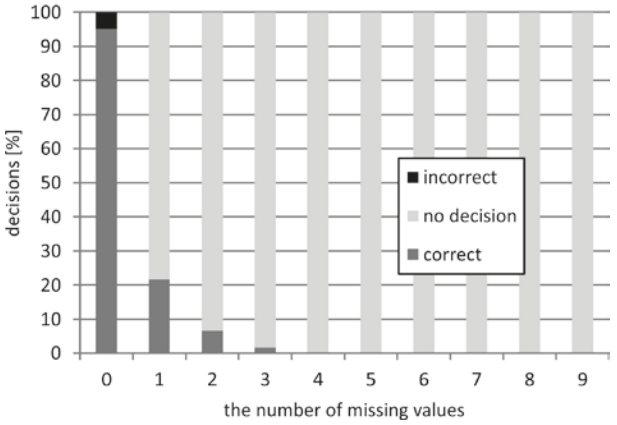

Figure 12. The results of Wisconsin Breast Cancer diagnostics using the rough support vector classifier — testing samples

Conclusions

The paper extends the idea presented in [10] to support vector machines. The rough support vector machine has been defined. It has been also explained how to derive and interpret the result of classification when the description of the object under classification represents a whole equivalence class. The results of experiments confirm that in the case of rough support vector machines, misclassification never results from the imperfection of the object description. It is a characteristic feature of all rough set–based classification systems. The experiments with various widths of input intervals have shown that the classifier is able to make the decision even with a high width. However, the acceptable width is different for particular samples. The results are similar to other types of rough set–based classification systems.

The high sensibility of the proposed system to missing values is undoubtedly an unusual surprise. It has also been confirmed by the tests with other data sets. In contrast to other rough set–based classification systems, the rough support vector machines do not classify the input data when even single missing values occur.

The future work on the rough support vector machines should focus on creating a classifier with imperfect data, i.e. selecting the support vector when the reference samples are described by intervals and contain missing values. It is already possible in the case of other rough set–based classification systems. The proposed system should be also tested as a part of an ensemble which is the main application of other systems in the group.

Acknowledgements

The project financed under the program of the Polish Minister of Science and Higher Education under the name ”Regional Initiative of Excellence” in the years 2019 - 2022 project number 020/RID/2018/19, the amount of financing 12,000,000 PLN.

References

[1] Clark, P.G., Grzymala-Busse, J.W.: Mining incomplete data with lost values and attribute-concept values. In: 2014 IEEE International Conference on Granular Computing (GrC), pp. 49–54 (2014). DOI 10.1109/GRC.2014.6982806

[2] Cortes, C., Vapnik, V.: Support-vector networks. Machine Learning 20(3), 273–297 (1995)

[3] Fisher, R.A.: The use of multiple measurements in

taxonomic problems. Annual Eugenics 7, 179–188 (1936)

[4] Hao, Z., Liu, B., Yang, X.: A comparision of multiclass support vector machine algorithms. In: 2006 International Conference on Machine Learning and Cybernetics, pp. 4221–4226 (2006). DOI 10.1109/ICMLC.2006.258947

[5] Hearst, M.A., Dumais, S.T., Osuna, E., Platt, J., Scholkopf, B.: Support vector machines. IEEE Intelligent Systems and their Applications 13(4), 18–28 (1998). DOI 10.1109/5254.708428

[6] Jin, B., Tang, Y., Zhang, Y.Q.: Support vector machines with genetic fuzzy feature transformation for biomedical data classification. Information Sciences 177(2), 476 – 489 (2007). DOI https://doi.org/10.1016/j.ins.2006.03.015

[7] Lingras, P., Butz, C.: Rough set based 1-v-1 and 1-v-r approaches to support vector machine multi-classification. Information Sciences 177(18), 3782 – 3798 (2007). DOI https://doi.org/10.1016/j.ins.2007.03.028

[8] Little, R.J.A., Rubin, D.B.: Statistical analysis with missing data, 2 edn. Wiley–Interscience (2002)

[9] Ma, Y., Guo, G. (eds.): Support Vector Machines Applications. Springer (2014)

[10] Nowicki, R.K.: Rough SetBased Classification Systems. Springer International Publishing, Cham (2019). DOI 10.1007/978-3-030-03895-3

[11] Pawlak, Z.: Information systems — theoretical foundations. Information Systems 6, 205–218 (1981)

[12] Pawlak, Z.: Rough sets. International Journal of Computer and Information Sciences 11(5), 341–356 (1982)

[13] Scholkopf, B., Kah-Kay Sung, Burges, C.J.C., Girosi, F., Niyogi, P., Poggio, T., Vapnik, V.: Comparing support vector machines with gaussian kernels to radial basis function classifiers. IEEE Transactions on Signal Processing 45(11), 2758–2765 (1997). DOI 10.1109/78.650102

[14] Song, H., Ding, Z., Guo, C., Li, Z., Xia, H.: Research on combination kernel function of support vector machine. In: 2008 International Conference on Computer Science and Software Engineering, vol. 1, pp. 838–841 (2008). DOI 10.1109/CSSE.2008.1231

[15] Vapnik, V.: Estimation of Dependences Based on Empirical Data. Springer-Verlag (1982)

[16] Vapnik, V.: Statistical Learning Theory. Wiley (1998)

[17] Weston, J., Watkins, C.: Multi-class support vector machines. In: Proceedings of ESANN99 (1999)

[18] Xiao, H., Sun, F., Liang, Y.: Support vector machine algorithm based on kernel hierarchical clustering for multiclass classification. In: 2010 International Conference on Electrical and Control Engineering, pp. 2201–2204 (2010). DOI 10.1109/iCECE.2010.542

[19] Zhang, J., Wang, Y.: A rough margin based support vector machine. Information Sciences 178(9), 2204 – 2214 (2008). DOI https://doi.org/10.1016/j.ins.2007.12.012

Robert K. Nowicki was born in Częstochowa, Poland, in 1972. He received the M.Sc. degree in electrical engineering from the Częstochowa University of Technology, in 1997, the Ph.D. degree in computer engineering from AGH University of Science and Technology, Kraków, Poland, in 2000 and the D.Sc. degree in computer engineering from the System Research Institute of the Polish Academy of Sciences, in 2010. Since 2010, he is an associate professor in the Department of Computational Intelligence at the Częstochowa University of Technology. He has published over 100 technical papers in journals, conference proceedings, and book chapters as well as 2 books. In 2010, he received the scientific prize of the Engineering Sciences Division of the Polish Academy of Sciences. His research interests include fuzzy systems, artificial neural networks, genetic algorithms, rough sets, and hybrid methods.

Konrad Grzanek, scientist, programmer and lecturer. Graduate of the Technical University of Łódź (FTIMS). Assistant professor at the Social Academy of Sciences. He holds a Ph.D. from Częstochowa University of Technology (CUT). His research interests focus on programming languages, software quality, software development processes, and artificial intelligence, in particular on combining machine learning methods with static software analysis. As a programmer, he is an advocate and promoter of the functional programming style. Author of over 30 publications related to various problems of computer science and software engineering.

Yoichi Hayashi received the Dr. Eng. degree in Systems Engineering from the Tokyo University of Science, Tokyo in 1984. In 1986, he joined the Computer Science department of Ibaraki University, Japan, as an Assistant Professor and was a Visiting Professor of Computer Science Department of the University of Alabama at Birmingham and the University of Canterbury, respectively. Since 1996, he is a full Professor of Computer Science department at Meiji University, Tokyo. He is the author of over 230 papers published in computer science. His current research interests include Artificial Intelligence, high performance classifier, Deep Learning, high accuracy rule extraction and medical informatics. He is the editorial board member of Artificial Intelligence in Medicine and was the Associate Editor of the IEEE Transactions on Fuzzy Systems, the Action Editor of Neural Networks. He is a senior member of the IEEE since 2000.