UDC 004.832:004.932.2 **doi: 10.26906/SUNZ.2026.2.184**

Svitlana Shapovalova, Olga Mazhara, Yurii Moskalenko, Vladyslav Titov

National Technical University of Ukraine "Igor Sikorsky Kyiv Polytechnic Institute", Kyiv, Ukraine

# RULE EXTRACTION FROM A KOHONEN SELF-ORGANISING MAP FOR EQUIPMENT CONDITION ASSESSMENT USING NOISY DIAGNOSTIC SIGNALS

**Abstract.** This paper proposes a method for extracting classification rules for one-dimensional (1D) diagnostic signals from a trained Kohonen self-organising map (SOM). To validate equipment condition assessment from time-series data, a classification problem involving second-order curves with similar fragments was formulated. A balanced dataset was generated from a mathematical model, the SOM was trained, and the corresponding clusters were identified using a multilayer perceptron (MLP). Rules were extracted using a decompositional approach in IF–THEN form, where conditions were defined in terms of the best-matching units (BMUs) of the input signal. The proposed approach enables robust classification of noisy signals. A software suite for populating a knowledge base with extracted rules was developed. Computational experiments on signal classification were conducted using both the SOM and a CLIPS-based rule-based system, with the number of antecedent conditions and the noise factor serving as simulation parameters. The results show that, with the maximum number of antecedent conditions, the classification accuracy of the rule-based system decreases by 1–3 percentage points depending on the noise factor.

**Keywords:** rule extraction, neural networks, Kohonen self-organising map, CLIPS, rule-based system, best-matching unit.

## Introduction

**Problem statement.** Neural networks have been widely adopted to solve applied artificial intelligence problems, including system identification, process control, decision-making, pattern recognition, data mining, and medical and technical diagnostics. However, a well-known limitation of this approach is the lack of interpretability mechanisms for the inference process, which is particularly critical in safety-sensitive systems.

One of the most common applied tasks in equipment condition monitoring is identifying operating conditions from time-series data. Real-time diagnostic systems must be computationally efficient while also providing explanations of their outputs. Extracting rules from trained neural networks is therefore both a timely and practically important research area.

**Analysis of recent research and publications sources.** A systematic review of rule extraction methods from feedforward neural networks, along with contemporary classification criteria for implementing explanations, is presented in [1].

Three main approaches to rule extraction from neural networks can be distinguished:

pedagogical – rule extraction for the network as a whole (the black-box principle);

decompositional – rule extraction for each individual component of the neural network (the white-box principle);

eclectic – a combination of the decompositional and pedagogical approaches.

In [2], an analysis and evaluation of the effectiveness of rule extraction algorithms across all three approaches using three datasets was conducted. Examples of rule extraction algorithms are presented in [3, 4] (pedagogical), [5, 6] (decompositional), and [7] (eclectic). A method for generating explanations of neural network inference is presented in [8].

For state identification from diagnostic signals, the decompositional approach was applied to the Kohonen self-organising map (SOM). A key property of the SOM is its ability to preserve topology by mapping high-dimensional input data onto a two-dimensional representation. Rule extraction from the SOM is discussed in [9] and [10]. In those studies, cluster identification within the SOM constituted a distinct extraction stage. In the present work, class labels (corresponding to equipment operating modes) are assigned to SOM clusters beforehand using a multilayer perceptron (MLP) for classification. This SOM–MLP neural network ensemble enables rapid diagnostics even from signals with missing data [9].

In addition, extracting classification rules from diagnostic signals raises the problem of overly large rule antecedents produced by the SOM. Addressing this requires a separate stage devoted to identifying the most informative signal features.

To formalise the rules derived from the mapping between diagnostic signals and SOM clusters, the CLIPS expert system language [12] was adopted.

**Task statement.** This paper proposes a method for extracting classification rules for 1D diagnostic signals from a trained Kohonen self-organising map.

To this end, the following objectives are pursued:

1) define the applied task of rule extraction;

2) determine the rule representation format;

3) develop a method for extracting classification rules from the SOM;

4) experimentally validate the effectiveness of the proposed method.

## The rule extraction task for second-order curve classification

The task was formulated as the extraction of rules from a self-organising map trained to classify time-series data corresponding to second-order curves with similar fragments, namely the upper arcs of a circle, an ellipse, and a parabola. To ensure maximum similarity, the parameters of the analytical equations for the second-order curves and their domains were carefully selected. The input signal to the neural network is a set of discrete function values defining the corresponding curve, with additive Gaussian noise. The mathematical model for generating training examples in the dataset is presented in [11]. Each example comprises the neural network input signal and the class (curve label) to which the signal corresponds.

Rule construction is based on matching each component of the input vector to the grid nodes of the Kohonen network. Under the unsupervised learning paradigm, the SOM requires no target vector: it learns to cluster the data from unlabelled examples. For each input signal, the best-matching unit (BMU) is computed – the grid node whose weight vector is closest to the input signal:

$$Dist = \sqrt{\sum_{i=1}^{i=n}(x_i - w_i)^2}, \qquad (1)$$

where $x_i$ is the i-th value of the input vector, $w_i$ is the value of the i-th weight of the BMU, and n is the length of the input vector.

Here, the Euclidean distance serves as the metric.

Extracting classification rules from the SOM requires:

1) a trained self-organising map;

2) a balanced dataset of input signal examples for rule construction;

3) a known mapping between classes and SOM clusters.

## Rule Representation

Rule extraction algorithms for neural networks can represent knowledge as mathematical expressions, symbolic logic expressions, fuzzy logic expressions, or decision trees. In practice, however, the two most common types of logical rules are:

IF–THEN (conjunctive): IF condition1 AND condition2 AND condition3 THEN RESULT;

M-of-N (subset selection): IF (M of the following N antecedents are TRUE) THEN RESULT.

Under the decompositional approach, the rule conditions directly reflect characteristics of the neural network.

An M-of-N rule treats its antecedent as true when M out of N Boolean expressions representing the network inputs are satisfied; the consequent is then interpreted from the neuron outputs. Notably, M-of-N rules can always be converted into IF–THEN form.

A more general approach to rule representation relies on first-order predicate logic (typically with certain restrictions). For example, in [13] the Gyan methodology is proposed, which encodes the knowledge of a trained network as restricted first-order predicate rules.

In this work, the extracted rules are formalised using the syntax of expert system shells to make them executable.

Among existing knowledge representation models, the production model aligns most closely with the IF–THEN rule format. Rule-based systems based on this model are commonly viewed as an adaptation of classical logic for artificial intelligence. Accordingly, the rules extracted from the SOM are expressed here in the CLIPS language [12]. CLIPS was chosen for its ability to construct antecedent logical expressions consistent with first-order logic, including existential (exists) and universal (forall) quantifiers, as well as for its support of basic mathematical and user-defined functions.

In the CLIPS environment, rule conditions are represented as fact templates. Facts can be either predefined (ordered) or generated dynamically during rule construction (unordered).

To simplify the generation procedure, rule antecedents may be built from unordered fact structures; however, preprocessing the data to identify significant structures improves rule readability.

For tasks with a well-defined output format (e.g., classification), output values are represented as ordered facts. A similar concept appears in the predicate-based rule generation approach of [13], where the target predicate maps to an ordered fact template in CLIPS.

## Method for extracting classification rules from SOM

Classification rules are extracted from the SOM in the following stages:

**Stage 1. Identifying the BMU for every input vector in the dataset.** All vectors from the dataset are fed to the trained network, and the BMU is determined for each:

$$\overline{X_j} \rightarrow BMU_k, if;$$
$$Dist_k = min(Dist_1, Dist_2, \ldots, Dist_m), \qquad (2)$$

where $\overline{X}_j$ is the j-th vector in the dataset, $BMU_k$ is the k-th SOM grid node that provides the best match for the input vector $\overline{X}_j$, $Dist_k$ is the distance between the vector $\overline{X}_j$ and the weight vector of the k-th SOM grid node (1), and m is the number of nodes in the SOM grid.

This yields a set of mappings $\overline{X}_j \rightarrow BMU_k$.

**Stage 2. Determining the range of input vector component values for each BMU.** Since a single BMU may correspond to several input vectors, all input signals mapped to each BMU are first identified:

$$\{\overline{X_1^k}, \overline{X_2^k}, ..., \overline{X_j^k}, ..., \overline{X_v^k}\}\text{-}{>}BMU_k, \qquad (3)$$

where $BMU_k$ is the BMU corresponding to the *k*-th grid node, belonging to the set of best-matching units identified in Stage 1; $\overline{X}_j^k$ is the j-th input vector from the set of vectors corresponding to the k-th BMU; and *v* is the number of input vectors corresponding to the *k*-th BMU.

All vectors on the left-hand side of (3) share the same dimensionality. The maximum and minimum values among all i-th elements must be found:

$$x_{ji\,min}^k = min(x_{j1}^k, x_{j2}^k, ..., x_{ji}^k, ..., x_{jn}^k), \qquad (4)$$

where $x_{ji}^k$ is the *i*-th element of the j-th input vector corresponding to the *k*-th BMU, and *n* is the length of the input signal.

The maximum value is determined analogously.

This yields the following set of value ranges:

$$\{(x_{j1\,min}^k, x_{j1\,max}^k), (x_{j2\,min}^k, x_{j2\,max}^k), ..., (x_{ji\,min}^k, x_{ji\,max}^k), ...,$$
$$(x_{jn\,min}^k, x_{jn\,max}^k)\}\text{-}{>}BMU_k, \qquad (5)$$

where $x_{ji}^k$ is the i – th element of the *j*-th input vector $\overline{X_j^k}$ corresponding to the *k*-th BMU, $BMU_k$ is the *k*-th SOM grid node that provides the best match for all vectors $X_j^k$ (6), n is the length of the input signal.

**Stage 3. Determining the classes for all BMUs identified in Stage 1.** At this stage, the right-hand side of (5) is replaced with the class label from the predefined class-to-cluster mapping:

$$BMU_k := A_r , \qquad (6)$$

where $A_r$ is the class identifier from the set SA.

The resulting set of mappings takes the form:

$$\{(x^k_{j1\ min}, x^k_{j1\ max}), (x^k_{j2\ min}, x^k_{j2\ max}),...,$$
$$(x^k_{ji\ min}, x^k_{ji\ max}),(x^k_{jn\ min}, x^k_{jn\ max})\}\text{->}A_r \qquad (7)$$

Expression (7) is a formal rule suitable for inclusion in an expert system knowledge base.

**Stage 4. Reducing the antecedent dimensionality.** The left-hand side of rule (8) has the same dimensionality as the input vector; consequently, high-dimensional input signals produce rules with a correspondingly large number of conditions. This increases the size of the knowledge base and slows down inference. It is therefore beneficial to reduce the rule antecedent at this stage by removing input signal fragments that are nearly identical across classes.

For the second-order curve classification task, regions were identified where the difference in curve values did not exceed a specified threshold ε (Fig. 1).

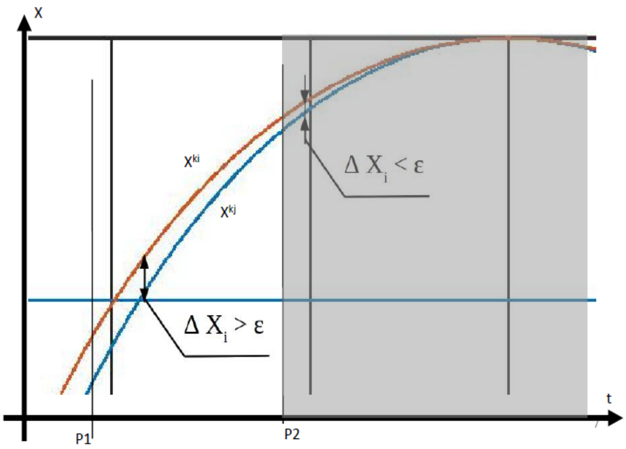


**Fig. 1.** Identification of curve regions for antecedent dimensionality reduction

The values $x_i$ falling within these regions were removed from all rules (8).

**Stage 5. Knowledge base rule representation.** In production form, rules (8) are written as:

**if**

$$x_1 \in [x^k_{j1\ min}, x^k_{j1\ max}]\ \&\ x_2 \in [x^k_{j2\ min}, x^k_{j2\ max}]\ \&\ ...\ \&\ x_{n^*} \in [x^k_{jn^*\ min}, x^k_{jn^*\ max}] \qquad (8)$$

**then**

where $x_i$ is the i-th value of the input vector;

$x_{ji}$ $min^k$ and $x_{ji}$ $max^k$ are the minimum and maximum values among all i-th elements of the input vectors;
$A_r$ is the class identifier from the set SA; and n* is the number of input vector values remaining after the removal of signal convergence regions (Stage 4).

**Stage 6. Knowledge base entry generation.** The knowledge base entries were generated in the CLIPS language following formula (9).

Fact template representation:

*(deftemplate figure*
$(slot\ x_1\ (type\ Float) * (range - 3.0\ 9.0))$
$(slot\ x_2\ (type\ Float) * (range - 3.0\ 9.0))$
$(slot\ x_3\ (type\ Float) * (range - 3.0\ 9.9))$
...
$(slot\ x_{100}\ (type\ Float) * (range - 3.0\ 9.9)))$

Example of an ellipse classification rule:

*(defrule ellipse*
$(figure\ (x_1\ ?x_1) * (x_2\ ?x_2) * (x_3\ ?x_3) \ldots (x_{50}\ ?x_{50})$
$(test\ (and\ (\geq ?x_1\ 2.277) * (\leq ?x_1\ 2.357)))$
$(test\ (and\ (\geq ?x_2\ 2.321) * (\leq ?x_2\ 2.401)))$
$(test\ (and\ (\geq ?x_3\ 2.342) * (\leq ?x_3\ 2.422)))$
...
$(test\ (and\ (\geq ?x_{50}\ 2.342) * (\leq ?x_{50}\ 2.422)))$
=>
$(assert\ (response\ ellipse))$
$(printout\ t$ The figure is ellipse" $clrf))$

## Computational experiments on rule extraction

Fig. 2 shows the components of the software system developed to extract classification rules for second-order curves from the Kohonen SOM under noisy input conditions, together with the intermediate data.

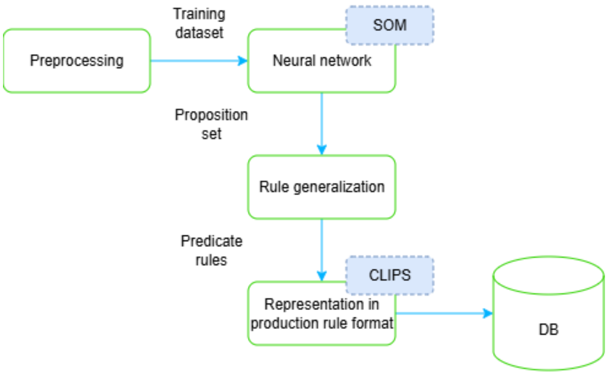


**Fig. 2.** Schematic diagram of rule extraction for the knowledge base

The neural network ensemble used for classification is described in [11]; it consists of the SOM (for clustering) and the MLP (for classification). Table 1 summarises the characteristics and corresponding values of the neural network, the training sample, and the training process.

Table 2 presents the conditions and results of the computational experiments. The computational experiments reveal that, compared with the neural network, the classification accuracy declines with increasing rule complexity (i.e., the number of antecedent conditions):

when 50 features remain after removing similar signal segments, accuracy drops by 1–2 percentage points;

for 40 remaining features, the drop ranges from 3 to 8 percentage points;

for 30 remaining features, the drop ranges from 3 to 10 percentage points.

Overall, the results demonstrate that rules produced by the proposed method yield rule-based systems whose inference accuracy is comparable to the neural network's.

*Table 1* – **Characteristics of the computational experiments**

| # | Category | Parameter | Value |
|---|---|---|---|
| 1 | Training sample parameters | Abscissa range | *[–3; 3]* |
| | | Ordinate range | *[0; 9]* |
| | | Noise | $\delta$ = *Rand(–x_max/k, x_max/k)* |
| 2 | Training characteristics | Number of SOM training examples | *100* |
| | | Number of SOM test examples | *100* |
| | | Number of MLP training iterations | *800* |
| 3 | Neural network characteristics | Input vector size | *100* |
| | | SOM grid dimensionality | *10×10* |
| | | Output vector size | *3* |

*Table 2* – **Results of the computational experiments**

| Noise factor δ | Classification accuracy, % | | | |
|---|---|---|---|---|
| | **Neural network** | **50 conditions** | **40 conditions** | **30 conditions** |
| 10 | 99 | 98 | 96 | 96 |
| 20 | 98 | 96 | 92 | 91 |
| 30 | 95 | 93 | 88 | 86 |
| 40 | 90 | 89 | 82 | 80 |

## Conclusions

1. To validate equipment condition assessment from time-series data, a classification problem involving second-order curves with similar fragments was formulated. A balanced dataset was generated from a mathematical model, a Kohonen self-organising map was trained, and the corresponding clusters were identified using an MLP.

2. Rules were extracted using a decompositional approach in IF–THEN form, with conditions defined in terms of the best-matching units (BMUs) of the input signal.

3. A method is proposed for extracting 1D signal classification rules from the output of a trained Kohonen SOM, enabling robust classification of noisy signals.

4. A software suite for populating a knowledge base with extracted rules was developed. Computational experiments on signal classification were conducted using both the SOM and a CLIPS-based rule-based system, with the number of antecedent conditions and the noise factor serving as simulation parameters. The results show that, with the maximum number of antecedent conditions, the classification accuracy of the rule-based system decreases by 1–3 percentage points depending on the noise factor.

Since most contemporary expert system tools employ a syntax similar to that of CLIPS, the proposed method is readily applicable to other knowledge bases and can be used to benchmark inference performance.

## Conflicts of interest

The authors declare that they have no conflicts of interest in relation to the current study, including financial, personal, authorship, or any other, that could affect the study, as well as the results reported in this paper.

## Use of artificial intelligence

The authors confirm that they did not use artificial intelligence technologies when creating the current work.

## References


1. Mekkaoui, S.E ., Benabbou, L., Berrado A. (2023). Rule-Extraction Methods From Feedforward Neural Networks: A Systematic Literature Review. arXiv preprint. https://doi.org/10.48550/arXiv.2312.12878
2. Augasta, M. G., Kathirvalavakumar, T. (2012). Rule extraction from neural networks — A comparative study. PRIME-2012, 404-408. https://doi.org/10.1109/ICPRIME.2012.6208380
3. Dattachaudhuri, A., Biswas, S.K., Chakraborty, M. et al. (2021). A transparent rule-based expert system using neural network. Soft Comput 25, 7731–7744. https://doi.org/10.1007/s00500-020-05547-7
4. Augasta, M.G., Kathirvalavakumar, T. (2012). Reverse Engineering the Neural Networks for Rule Extraction in Classification Problems. Neural Process Lett 35, 131–150. https://doi.org/10.1007/s11063-011-9207-8
5. Zilke, J.R., Loza Mencía, E., Janssen, F. (2016). DeepRED – Rule Extraction from Deep Neural Networks. In: Calders, T., Ceci, M., Malerba, D. (eds) Discovery Science. DS 2016. Lecture Notes in Computer Science. 9956. Springer, Cham. https://doi.org/10.1007/978-3-319-46307-0_29
6. Hongjun Lu, Setiono R., Liu H. (1996). Effective data mining using neural networks. IEEE Transactions on Knowledge and Data Engineering. vol. 8. no. 6, 957-961. https://doi.org/10.1109/69.553163

7. Iqbal R. (2011). Eclectic Extraction of Propositional Rules from Neural Networks. ICCIT 2011, 234-239. https://doi.org/10.1109/ICCITechn.2011.6164790
8. Guidotti, R., Monreale, A., Ruggieri, S., Pedreschi, D., Turini, F., Giannotti, F. (2018). Local Rule-Based Explanations of Black Box Decision Systems. arXiv. https://doi.org/10.48550/arXiv.1805.10820
9. Hammer, B., Rechtien, A., Strickert, M., Villmann, T. (2002). Rule Extraction from Self-Organizing Networks. In: Dorronsoro, J.R. (eds) Artificial Neural Networks — ICANN 2002. Lecture Notes in Computer Science, 2415. Springer, Berlin, Heidelberg. https://doi.org/10.1007/3-540-46084-5_142
10. Malone, J., Mcgarry, K., Wermter, S., Bowerman, C. (2006). Data mining using rule extraction from Kohonen self-organising maps. Neural Computing and Applications. 15. 9-17. https://doi.org/10.1007/s00521-005-0002-1
11. Shapovalova S., Moskalenko Yu. (2019). Increasing the share of correct clustering of characteristic signal with random losses in self-organizing maps. Eastern-European Journal of Enterprise Technologies. 2. 4 (98). 13-21. DOI: https://doi.org/10.15587/1729-4061.2019.160670
12. Riley, G. (2022). Adventures in Rule-Based Programming: A CLIPS Tutorial. Kindle Edition, 200. https://clipsrules.net/airbp.html
13. Nayak R. (2009). Generating rules with predicates, terms and variables from the pruned neural networks. Neural Networks. 22, 4, 405–414. https://doi.org/10.1016/j.neunet.2009.02.001.

Received (Надійшла) 18.01.2026
Accepted for publication (Прийнята до друку) 29.04.2026
Publication date (Дата публікації) 22.05.2026

ABOUT THE AUTHORS / ВІДОМОСТІ ПРО АВТОРІВ

**Шаповалова Світлана Ігорівна** – кандидат технічних наук, доцент, доцент кафедри автоматизації проектування енергетичних процесів і систем, Національний технічний університет України «Київський політехнічний інститут імені Ігоря Сікорського», Київ, Україна;
**Svitlana Shapovalova** – Candidate of Technical Sciences, Associate Professor, Associate Professor of Department Digital Technologies in Energy, National Technical University of Ukraine "Igor Sikorsky Kyiv Polytechnic Institute", Kyiv, Ukraine;
e-mail: lanashape@gmail.com; ORCID Author ID: http://orcid.org/0000-0002-3431-5639;
Scopus Author ID: https://www.scopus.com/authid/detail.uri?authorId=35367847700.

**Мажара Ольга Олександрівна** – кандидат технічних наук, доцент кафедри автоматизації проектування енергетичних процесів і систем, Національний технічний університет України «Київський політехнічний інститут імені Ігоря Сікорського», Київ, Україна;
**Olga Mazhara** – Candidate of Technical Sciences, Associate Professor of Department Digital Technologies in Energy, National Technical University of Ukraine "Igor Sikorsky Kyiv Polytechnic Institute", Kyiv, Ukraine;
e-mail: yuramuv@gmail.com; ORCID Author ID: https://orcid.org/0000-0001-7887-6764;
Scopus Author ID: https://www.scopus.com/authid/detail.uri?authorId=57190385872.

**Москаленко Юрій Володимирович** – доктор філософії, старший викладач кафедри автоматизації проектування енергетичних процесів і систем, Національний технічний університет України «Київський політехнічний інститут імені Ігоря Сікорського», Київ, Україна;
**Yurii Moskalenko** – PhD, Senior Lecturer of the Department Digital Technologies in Energy, National Technical University of Ukraine "Igor Sikorsky Kyiv Polytechnic Institute", Kyiv, Ukraine;
e-mail: yuramuv@gmail.com; ORCID Author ID: https://orcid.org/0000-0002-0824-9201;
Scopus Author ID: https://www.scopus.com/authid/detail.uri?authorId=57210336448.

**Тітов Владислав Миколайович** – аспірант, Національний технічний університет України «Київський політехнічний інститут імені Ігоря Сікорського», Київ, Україна;
**Vladyslav Titov** – Post-Graduate Student, National Technical University of Ukraine "Igor Sikorsky Kyiv Polytechnic Institute", Kyiv, Ukraine;
e-mail: 19Titov.vlad.Titov19@gmail.com; ORCID Author ID: https://orcid.org/0009-0000-5780-5596;
Scopus Author ID: https://www.scopus.com/authid/detail.uri?authorId=59729089700.

## Екстракція правил із самоорганізувальної карти Кохонена для визначення стану обладнання за зашумленим діагностичним сигналом

С. І. Шаповалова, О. О. Мажара, Ю. В. Москаленко, В. М. Тітов

**Анотація.** Об'єктом дослідження статті є методи екстракції правил з нейронних мереж. Метою статті є представлення методу екстракції правил класифікації 1D діагностичних сигналів на основі навченої нейронної мережі Кохонена. Conclusions: для тестування задачі визначення стану обладнання за часовим рядом показників поставлено задачу класифікації кривих другого порядку за подібними фрагментами. За математичною моделлю створено збалансований датасет, проведено навчання Kohonen self-organising map, за допомогою MLP визначено відповідні кластери; правила екстракції визначалися за декомпозиційним підходом у форматі IF-THEN, де умови визначалися на основі одиниць найкращої відповідності вхідного сигналу мережі BMU (Best Matching Unit); запропоновано метод екстракції правил класифікації 1D сигналів на основі результатів класифікації навченої нейронної мережі Кохонена, який дозволяє класифікувати зашумлені сигнали; розроблено програмний комплекс для екстракції правил у базу знань. Проведено обчислювальні експерименти з класифікації сигналів SOM та Rule based system на основі CLIPS, параметрами моделювання яких була кількість умов в правилах бази знань та коефіцієнт шуму сигналу. Визначено за максимальної розмірності умовної частини в залежності від рівня шуму точність класифікації на Rule based system знижується від 1 до 3%.

**Ключові слова:** екстракція правил, нейронні мережі, Kohonen self-organising map, CLIPS, Rule based system, Best Matching Unit.