UDC 004.67:616-71
DOI: 10.15587/1729-4061.2024.298805

APPLICATION OF A DATA STRATIFICATION APPROACH IN COMPUTER MEDICAL MONITORING SYSTEMS

Volodymyr Donets
Corresponding author
Postgraduate Student*
E-mail: vol.donets@gmail.com

Dmytro Shevchenko
Postgraduate Student*

Maksym Holikov*

Viktoriia Strilets
PhD*

Serhiy Shmatkov
Doctor of Technical Sciences*

*Department of Theoretical and Applied Systems Engineering
V. N. Karazin Kharkiv National University
Svobody sq., 4, Kharkiv, Ukraine, 61022

The research object is the processes occurring in the data stratification subsystem in the medical monitoring computer system, which is part of the decision support system. Such a subsystem aims to solve data analysis and processing problems in the medical monitoring system. Among them, the problems of anomaly detection, data marking, state determination, selection of the most informative variables, and justification of decision-making are selected for solving.

The paper proposes the structure and implementation of the data stratification subsystem in the decision support system. The subsystem contains modules for anomaly detection and an autoencoder, a clustering module using an advanced multi-agent clustering method, and a state detection module with a modified neural network training procedure.

Modules of the stratification subsystem have been tested using diabetes monitoring data. The results showed that the clustering module provides 25.7 % lower accuracy than the achieved neural network. The accuracy difference is explained by the complexity of the data and the lack of adaptability of the proposed method to solving such problems. It is shown that the method of determining the overall informativeness of variables covers 90 % informativeness with 10 variables, comparable to the variability data. In general, the flexible nature of the proposed stratification subsystem allows for solving the problems.

The proposed stratification subsystem offers a robust solution for improving treatment strategies and decision-making in computer medical monitoring systems. Its versatility allows it to be used in any system where decision support is needed, providing valuable information about informative variables and decision-making features for clinicians and researchers

Keywords: data stratification, anomaly detection, fuzzy clustering, neural network, sensitivity analysis

Received date 21.02.2024
Accepted date 29.03.2024
Published date 30.04.2024

How to Cite: *Donets, V., Shevchenko, D., Holikov, M., Strilets, V., Shmatkov, S. (2024). Application of a data stratification approach in computer medical monitoring systems. Eastern-European Journal of Enterprise Technologies, 2 (9 (128)), 6–16. https://doi.org/10.15587/1729-4061.2024.298805*

1. Introduction

An advanced computerized medical monitoring system's crucial goal is to determine the appropriate treatment strategy with the best possible outcome by analyzing the collected patient data. These systems usually generate a lot of stochastic data [1]. The data obtained by such systems allow identification of the patient's condition but require predefined logic or expert (medical) analysis [2]. Machine learning techniques enable automatic data analysis in such systems with more profound data analysis logic that may be possible with expert interaction. These methods will make it possible to study the relationship between the data and the patient's condition, which leads to an increase in the quality of treatment [2]. Automated medical monitoring systems are used as decision support systems for disaster prevention [1, 2]. Existing information systems partially solve the problem of element stratification in computer medical monitoring systems when clustering data for determining conditions, classifying conditions, and justifying decision-making regarding the classification of conditions [3].

Research on this topic is essential to ensure the effective use of machine learning for analyzing and classifying medical data. In practice, this will allow automation of the patient data stratification process, which in turn will increase the accuracy and speed of decision-making in medical practice. This will also contribute to implementing optimal treatment planning systems and improving the quality of medical services.

2. Literature review and problem statement

The works [1, 2] consider the results of developing patient monitoring systems using the Internet of Things. These systems are introduced in the field of providing medical services. It is shown that the advantage of such systems is the possibility of remote monitoring of patient's conditions. However, the issue of decision-making by such systems regarding the treatment strategy or the patient's condition determination, which, with many patients, requires many doctors' involvement, remained a drawback. This is due to the introduction of excessive complexity in developing such systems, which makes such studies impractical. The solution to this problem may be the development of a separate decision-making system that can be easily implemented in monitoring systems.

The paper [3] considers implementing machine learning methods in the limbs medical monitoring system. Such systems are used in the field of prosthetics services. The advantages of machine learning methods for determining the most informative variables in electromyography data are shown. However, the disadvantage is that the data considered in such a system have only a physical nature. This is due to the method of obtaining data and the narrow range of the developed system applications.

The study [4] considered big data analysis in the field of medical monitoring and indicated the problems of using such data by healthcare experts for treatment. It is shown that patient data can be used to improve the quality of treatment. However, the sheer amount of data generated by medical devices can overwhelm healthcare professionals, making it challenging to extract meaningful information. In addition, system incompatibilities can impede data sharing and create fragmented patient records. This is explained by the standards inconsistency of the systems designed to process medical data and the need for doctors to analyze more information than they can process effectively. The solution to this problem can be implementing methods for determining informative variables in a big data stream.

In [5, 6], the problem of the propensity of machine learne ing models and methods to bias in medical data analysis is considered. The research was conducted at the intersection of language models and medical data processing. It is shown that the models can use certain state variables too biasedly, which have no real influence on the outcome of the disease courses. The study's advantage is that it provides an overview of advanced generative language models and their biases in medical data analysis problems and provides practices to avoid such situations. However, the research lacks an automatic methods overview that can detect dependencies and show specialists in which cases the models make biased decisions. This is due to the need to show the problem of bias inherent in machine learning methods and models. The solution to this problem may be applying informativeness analysis to specific data and combining its results with defined bias variables.

The works [7, 8] consider the medical device's affordabile ity in the field of providing medical services. The advantage of the studies is that they analyze the pricing policy of a wide range of medical devices. It is shown that medical devices require complex certification and testing, which increases their cost tenfold. However, the shortcoming of the work is the lack of an overview of methods for solving the problem of the high cost of medical devices and ways to reduce their cost. This is related to the work's purpose of reviewing the pricing policy for medical devices.

The works [9, 10] consider testing methods and models for modules of cluster analysis and data classification in the field of medical monitoring. The advantage is that it shows how the proposed methods and models can improve the accuracy of healthcare professionals' work and healthcare quality. The research disadvantage is the lack of analysis of the combination of these methods in the context of medical monitoring data. This is because the work aimed to check the accuracy of the proposed methods.

For the reasons discussed above, the concept of a data stratification subsystem was proposed in [11]. The work [12] considers data stratification as a data analysis methodology in the country's economic development field. It is shown that such a methodology includes applying methods for anomaly detection, autoencoder, cluster analysis modules, classification, and determination of overall informativeness (to reduce processed data and argue decisions in such a system). The advantage is that it separates the stratification concept and the quality control of its functioning in economic monitoring systems. However, the drawback of the study is the lack of performance testing of the proposed subsystem in other areas. This is due to the focus of the work on economic data analysis and a different set of technologies. The solution to this problem may be the introduction of a stratification subsystem for the analysis of medical monitoring data with the involvement of new methods for determining informativeness.

The disadvantages of existing solutions in medical monitoring systems are the absence or narrow specificity of the implementation of data analysis methods and models, the high cost of existing solutions, and the complexity of justifying the decisions made. Using the stratification approach considered in [11, 12] in the computer medical monitoring system will help simplify analyzing medical monitoring data. Namely, it will help to increase the accuracy of state detection and the accuracy of the justification of such detection, that is, the current informativeness determination.

3. The aim and objectives of the study

The aim of the study is to increase the accuracy of the detection of patients' conditions and the accuracy of the justification of such detection using the data stratification subsystem in a computer medical monitoring system for the accuracy of the justification of decision-making. This will make it possible to develop a data stratification system that will increase the reliability of determining conditions, simplify decision-making on the condition of patients, and justify the decisions taken by medical professionals.

To achieve the aim, the following objectives were set:

– to develop modules for anomaly detection, autoencoder, clustering, classification, and analysis of general and current informativeness and integrate them into an integrated stratification subsystem;

– to check the effectiveness and accuracy of the proposed modules of the stratification subsystem using actual medical data.

4. Materials and methods

The study's object is the processes occurring in the data stratification subsystem in a computer medical monitoring system. The subsystem includes modules for clustering, classification, and determining variables' overall and current informativeness.

The central hypothesis of the work is that the consistent use of the stratification subsystem modules can increase the efficiency of determining and justifying the patient's current conditions in a computer medical monitoring system.

Usually, the possible states of patients in computerized medical monitoring systems are not defined, so a decision support system architecture (Fig. 1) [11] was proposed, which consists of decision support modules and a stratification subsystem (Fig. 1, green blocks). The context diagram shown in Fig. 1 is an adaptation of the stratification subsystem to a computer medical monitoring system. The stratification subsystem in the proposed implementation contains a Cluster Analysis (CA) module based on a modified multi-agent fuzzy clustering method [9]. The system also includes a Neural Network Classifier (NNC) module, based on a modified neural network learning procedure [10], and a Sensitivity Analysis (SA) module, which will be discussed in more detail in this work.

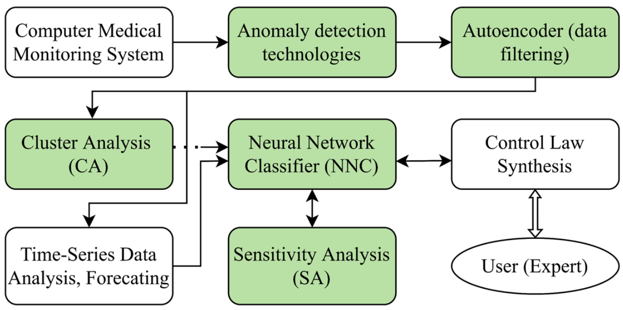


Fig. 1. Context diagram of the computer medical monitoring system implementation

Data from the computer medical monitoring system are analyzed for abnormal values resulting from errors in sensors, networks, or algorithms for obtaining these data [13]. In addition, the autoencoder can be used to understand the nature of the processed data further, indicate the value of the anomaly, and compress the data if necessary due to the general capabilities of the system [14]. Selected anomaly values are rejected. The processed (if necessary) and filtered data is sent to the cluster analysis module, which allows the system to identify target clusters and the number of them that can be determined in the collected data. The selected clusters are classes of conditions (or simply conditions) where the data of the computerized medical monitoring system are currently located. Cluster data is used to label the data stream with cluster labels. The observed data are used to train an Artificial Neural Network (ANN) model (Fig. 1, NNC module) to classify the data, and the trained ANN model is used to classify the patient's condition and evaluate the predicted data, which is used to synthesize the control law. The SA module was introduced to improve the argumentation of the NNC module. The SA module can also perform another task of selecting the most valuable variables.

The stratification subsystem implementation of the computer medical monitoring system can be carried out in several modes:

– fully dynamic (in uninformed learning mode) means that the stratification subsystem does not have training data and experts do not label the input data. The subsystem manually determines the number of clusters and labels data using the CA module. The clustering result may change with more data collected, which often leads to retraining and reselection of the NNC module architecture and, consequently, to a change in the informativeness score;

– partially dynamic (in partly informed mode), experts determine the number of clusters in advance, so the NNC module does not need changes in its architecture. All system updates occur if the data is updated. Accordingly, the informativeness values updates are based on the updates of the data labels received from the CA module;

– deterministic (informed learning mode): experts label the initial training dataset according to the corresponding mode. Thus, the CA and NNC modules are trained only once. This mode is suitable for general testing or tuning of the stratification subsystem.

Considering this, the deterministic mode was chosen for the stratification subsystem operation. It also simplifies testing methods for obtaining informativeness coefficients. The NNC module can be used to check the accuracy of the CA module because the ANN model inside the NNC module can be considered a universal data approximator. If the accuracy of the results on the NNC module is comparable to that obtained using the CA module, this indicates its effectiveness. Analysis of variance can be applied to test the SA module. Suppose the determined percentage of variability and the percentage of informativeness have the same number of variables. In that case, this indicates the effectiveness of the overall variable informativeness method. The verification of current informativeness should include the general results, and the appearance of the same variables should also indicate the method's effectiveness. Therefore, the study assumes that the clustering model has sufficient complexity to cover medical monitoring data. It allows the ANN model to be more complex than the clustering model. It is also assumed that the proposed methods of determining overall and current informativeness can be evaluated using other methods (using a third-party model).

To implement the proposed stratification subsystem, software with the Python language interpreter and the Jupyter Notebook runtime environment were used. In more detail, mathematical and scientific libraries were used for the following tasks:

1. Data processing, preparation, and analysis: Numpy, Pandas, ScikitLearn.

2. Implementation of ANN models, autoencoder, and methods for determining informativeness: TensorFlow.

3. Visualization of work results: Matplotlib, Seaborn.

The developed software can be run on any computer that supports the Python 3.11 interpreter. However, to speed up the work, the Nvidia CUDA library was used to accelerate calculations by using the video adapter, the support of which is integrated into the TensorFlow library.

5. Results of research on the processes occurring in the data stratification subsystem

5. 1. Modules development and their integration into the stratification subsystem

Anomaly detection technologies.

Typical data processing practices involve using mathematical models, methods, and technologies to detect outliers before submitting data for computation. Data preprocessing is important because outliers often indicate certain phenomena, errors, or deficiencies in data collection. For example, in medical systems, abnormal values can be collected due to errors in entering indicators, equipment errors, or patients' non-compliance with certain conditions when passing laboratory tests. Thus, there is a need to identify and remove outliers from a dataset.

First of all, it is worth identifying abnormal values and outliers. These two terms often describe anomalous instances of data, sometimes interchangeably. Anomalies or outliers are cases that differ and fall outside the normal range of the data distribution.

After considering the advantages of using mathematical models, methods, and technologies for detecting outliers, it was determined that most mathematical models and methods work well on normally distributed data. In turn, detecting and removing outliers helps to reduce the impact of outliers on the data distribution and eliminate skewed data. Thus, in the problem of data clustering when using centroid-based methods, removing outliers provides better clustering because there is no displacement of the cluster centroid due to the inclusion of outliers in the cluster. This is the main advantage of anomaly detection in tasks where datasets are processed for further use. An additional advantage may be detecting abnormal values for specific tasks, such as detecting abnormal activity or determining a change in a patient's condition by analyzing his indicators.

Mathematical models and methods for detecting anomalous values that best manifest themselves were defined in the study [13]. These are isolation forests and generative competitive networks. However, autoencoders have also been shown to perform relatively well in anomaly detection.

The created anomaly detection module uses traditional mathematical models and anomaly detection methods as a combination of several for better data processing. The following methods were combined:

– Z-score. This method is also called standard observational assessment. It indicates the location of the initial estimate in terms of the mean value when measured in units of standard deviation [15];

– insulating forest. This mathematical model uses binary decision trees. The basic idea of an isolated forest is that emissions are rare and far from other observations [16].

The isolated forest method was first applied to detect outliers in the dataset. The basic concept remains the same: the data space was partitioned into subspaces by choosing random partitions and determining the depth of each instance. Observations that have shorter paths are considered anomalous.

A Z-score was then used to detect additional outliers. The mean (μ) and standard deviation (σ) were calculated for each variable in the data. Observations with a high Z-score are considered outliers. Thus, the structural data of the isolated forest were combined with the information on the distance from the average value of the Z-score. The final indicator is calculated according to the following formula:

$$Score_i = IF_{score_i} * \frac{1}{1+e^{-z}}, \quad (1)$$

where IF_{Score_i} is the isolated forest data depth representation, z is the Z-score value. This expression uses the scalar product of the IF_{Score} and the inverse sigmoid function of the Z-score for each sample. This made it possible to increase the efficiency of detecting anomalies in the dataset using this method, which combined a mathematical model and a method for detecting anomalies, where one is based on the data structure and the other on the variables' statistical characteristics.

Autoencoder.

This module uses an autoencoder for several key areas, such as dimensionality reduction and data filtering through anomaly detection. A comparison [14] was made between standard and variational autoencoders for the data dimensionality reduction problem, and it was determined that the standard autoencoder performs better. Therefore, in this module, it is used for both tasks.

An autoencoder is a feed-forward neural network where the input data matches the output data. This type of network compresses the input data to a bottleneck represented by low-dimensional data and then reproduces the output data from this compressed representation. A bottleneck is a compact union or generalized input data compression, also called a hidden space representation.

When considering the autoencoder architecture, three main parts can be distinguished: the encoder, the bottleneck, and the decoder. Since autoencoders are currently being modified, a new architecture was created (Fig. 2).

The autoencoder's architecture is quite versatile because specific hyperparameters, such as the sizes of layers and activation functions, can be changed.

An autoencoder has several types of neural network layers, such as:

– Linear. This layer uses a linear transformation for the input data $y = xA^T + b$, where A^T and b are weights that are adjusted during training; x – input data;

– Activation. Defines the activation function that is applied to the data;

– Dropout. During training, this layer randomly resets some input data elements to zero with a given probability, using samples from a Bernoulli distribution. The effectiveness of this method of regularization and prevention of neuron co-adaptation has been confirmed as described in [17];

– BatchNorm. It uses batch normalization for input data with dimension 2 (as an example, dimension 16×128), as discussed in [18].

The parameters and layers of the autoencoder are considered in more detail in [14], where the dimensions of the hidden layers and additional parameters used in this module are defined.

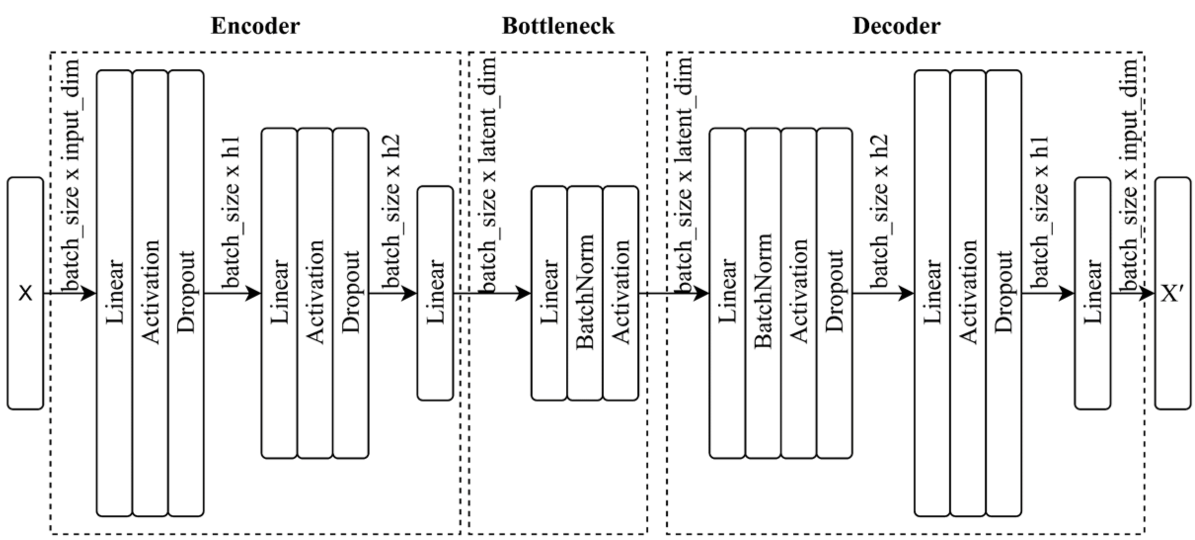


Fig. 2. Autoencoder architecture

As a result, the above autoencoder architecture has advantages such as robustness to retraining and improved gradient descent by batch normalization.

Considering the tasks that the autoencoder must perform, it was determined how they can be solved:

1. The problem of dimensionality reduction assumes that the input data will be compressed to a given size while preserving its basic properties. This was achieved by feeding the input data to the autoencoder and saving the processed data from the bottleneck.

2. The data filtering task is based on anomaly detection. The input data was given to the autoencoder and compared with the output to determine whether a value was anomalous. For normal values, the input and output were approximately the same, while for anomalies, the input and output differed significantly, allowing outliers to be identified. The same metric used for training was used to compare the data, namely mean square error (MSE).

It should be noted that before using the autoencoder, it should be trained on data from the same distribution as the target data. Otherwise, the autoencoder will not work correctly. Another essential property of an autoencoder is that it does not require correct labels for training, which defines it as a mathematical model that uses unsupervised learning. This type of training makes it easy to apply the autoencoder to any tabular data.

Cluster analysis module.

The cluster analysis module uses the multi-agent fuzzy clustering method [9]. The method was obtained by combining the modified c-means and multi-agent methods [9]. For the proposed method of multi-agent fuzzy clustering, the labeling was introduced: $X=\{x_n\}$ are agents that are entries in the input data, $C=\{c_j\}$ are agents representing cluster centers, and $X_j=\{x_{ji}\}$, x_{ji} in X are agents, representing cluster elements.

The problem of clustering involves finding $\{C, X_j\}$ – the target number of clusters and the distribution of input data elements between specific clusters. From the point of view of the multi-agent approach, this means the formation of cluster agents C with their agent elements X_j.

An essential step in determining the clustering method is the choice of a metric for calculating the distance between elements. The best results were obtained for the Kullback-Leibler asymmetric entropy [9], and the formula for calculating the intra-cluster distance is as follows:

$$M(c_k,H_k)=\frac{1}{P_k}\sum_{x_n}^{H_k} d(x_n,c_k), \quad (2)$$

where P_k is the number of elements in the k-th cluster, H_k is the elements set of the k-th cluster, and $d(x_n, c_k)$ is the distance, according to the given metric, between the center of the k-th cluster and the n-th element of this cluster.

The loss function was determined by using expression (2):

$$LOSS(X,C,H)=\frac{1}{K}\sum_{k=1}^{K} M(c_k,H_k). \quad (3)$$

According to the classical method of fuzzy clustering [19], cluster centers are optimized according to the formula:

$$c_k \leftarrow \left[\sum_{i=1}^{P_k} w_{ik}x_i\right] / \left[\sum_{i=1}^{P_k} w_{ik}\right], \quad (4)$$

where is the expression to calculate the membership matrix adjustment:

$$w_{nk}=p(x_{nk},c_k) / \left[\sum_{i=1}^{P_k} p(x_{ik},c_k)\right], \quad (5)$$

where $p(x_{nk},c_k)=(pi\times h[1+MD^2(x_{nk},c_k)/h^2])$ is the distribution density at the point x_{nk} for the cluster c_k, determined according to the Cauchy law using the normalization parameter h.

Based on the introduced definitions, the algorithm of the multi-agent fuzzy clustering method can be described as follows [9]:

1. Randomly determine the centers of agent clusters C that exceed the target number of clusters determined by experts or in some ratio to the number of agent elements X. Set a critical number of elements in each cluster that usually does not exceed 50 % of the $|X|/|C|$ value.

2. Using the Kullback-Leibler metric, select the cluster elements X_j closest to each cluster to create agent clusters and calculate the membership value using the formula (5).

3. Configure agent clusters C according to formula (4) and redefine agent elements X_j for each cluster.

4. Calculate the value of the intra-cluster distance for each cluster agent and its element agents according to formula (2) and calculate the value of the loss function according to formula (3).

5. If the cost function value has decreased more than the minimum step, return to step 2. In addition, if the method finds the number of clusters according to a specific value of the cost function, stop the execution when this condition is reached.

6. If the number of clusters has not reached the minimum, delete the cluster with the largest intra-cluster distance and go to step 2. Otherwise, stop execution.

Neural network classifier module.

The NNC module is based on a typical fully connected neural network model. This model has a Softmax function (or multiple logistic regression) to activate the outputs [10], a common choice for models that solve the data classification problem [20, 21]. A Sigmoid activation function is used for the intermediate layers, providing a smooth non-linearity in the behavior of the model. Other activation functions such as ReLU, Tanh, and other specialized functions can be applied to intermediate levels if necessary.

The ANN model is designed to solve the classification problem. Therefore, the categorical cross-entropy is used as a loss function [10, 22]. The peculiarity of this model is that the number of layers and neurons inside is dynamically determined by the procedure of accelerated learning and the procedure of fitting hyperparameters, described in detail in [10]. This fact distinguishes the built model from other modern architectures, making it more flexible and adaptable to the tasks that this module solves. Using the proposed procedures allows for avoiding the occurrence of incorrect classification values in case of significant errors in the input data [10].

The procedure for adjusting model hyperparameters uses an improved method of estimating parameters of structural and parametric models of systems and processes. This method is based on using ANNs trained by the stochastic approximation method. In addition, an adaptive algorithm is used to synthesize solutions with a delay. Parameters adjustment is carried out using adaptive computer control based on the principle of minimum disturbance, using spring and conjugate gradient methods and simulation of the movement of bee colonies [10]. When comparing the models obtained during training, the change in the dispersion of the signal is estimated, which determines the stability of a particular model. The most stable model is the primary model in the NNC module.

The procedure for accelerated ANN learning is based on a two-stage hybrid algorithm. Repeating these steps leads to fast learning of the network [10]:

1. Selection of linear network parameters by the pseudo-inversion method.

2. Optimization of nonlinear parameters of activation functions (window centers and widths).

Sensitivity analysis module.

Two methods are used to evaluate the informativeness of the input data values, which are designed to determine current and overall informativeness. The current variable informativeness can be used to determine the reasons for the decisions made by the NNC module and thereby help the system's end user better justify the decisions made. Evaluation of the overall variable informativeness performs two functions. The first occurred when reducing the total number of variables analyzing the stratification subsystem of the computer medical monitoring system (variable selection task). The second is the most significant impact of variable data for more careful monitoring in the system (for example, for display in the window of a running program).

The overall informativeness gradient method of the sensitivity analysis module.

The main feature of the developed method is that the neural network model is used to select the most informative variables, taking into account their nonlinear relationships. Reducing the number of input variables by finding a set of the most informative input variables becomes possible if necessary. The definition of the method for evaluating the informativeness of variable input data is given below. For this, S_B is defined as a set of the most informative variables; the volume of this set can be defined as constant or depends on some level of total informativeness (for example, the preservation of S_B information). Then, S_B is in S, where S is the set of output-input variables. Then $S=\{s_n\}$, $n=\left(1,N\right)$ terminates as a Taylor series preserving only extreme terms. This helps to determine the output variance of the ANN model as arbitrary linear functions of the vector output:

$$D_{Y_k}=\left(deltaY_k\right)^T M_S deltaY_k= \\ =\sum_{n=1}^{N}\left(\frac{dY_k}{ds_n}\right)^2 q_{s_n}^2+\sum_{n=1}^{N}\sum_{m=1,n\neq m}^{N} r_{nm}\frac{dY_k}{ds_n}\frac{dY_k}{ds_m}q_{s_n}^2 q_{s_m}^2, \quad (6)$$

where M_S is the covariance matrix of the input variables, q_{s_n} is the standard deviation of the n-th variable, r_{nm} is the correlation value between the n-th and m-th variables, $Y_k(S)$ is a linear function that represents the k-th output of the trained ANN model, with $k=\left(1,K\right)$ K is the number of expected states/classes.

To investigate the actual ANN model, the functions representing the outputs and inputs of the ANN model were defined as F_k^{out} and F_n^{in}, respectively. The dispersion of ANN outputs, taking into account its inputs, is defined in [10] using the following equation:

$$D_{F_k^{out}|F_n^{in}}=\left(\frac{dF_k^{out}}{dF_n^{in}}\right)^2 q_{dF_n^{in}}^2+ \\ +\left(\sum_{t=1,t!=1}^{N} r_{nt}\frac{dF_k^{out}}{dF_n^{in}}q_{dF_t^{in}}\right)\frac{dF_k^{out}}{dF_t^{in}}q_{dF_n^{in}}, \quad (7)$$

where for the estimation of $q_{dF_n^{in}}$, the gradients of the ANN model were calculated on all available labeled data, taking into account the categorical cross-entropy loss.

Taking this into account and determining the signal energy in [10], the energy signal from each ANN output can be described as:

$$E_k=\sum_{n=1}^{N}\left|D_{F_k^{out}|F_n^{in}}\right|. \quad (8)$$

Then, the coefficient of influence of F_n^{in} on F_k^{out} can be determined by the expression:

$$B_{kn}=\frac{\left|D_{F_k^{out}|F_n^{in}}\right|}{E_k}. \quad (9)$$

Considering the influence coefficient, the informativeness value for each variable is determined using the following expression:

$$GBI_n=\left(\sum_{k=1}^{K}B_{kn}\right)\Big/\left(\sum_{n=1}^{N}\sum_{k=1}^{K}B_{kn}\right). \quad (10)$$

The current informativeness integrated gradients of the sensitivity analysis module.

Informativeness values obtained using the gradient method better reflect the effect of each variable using only all available labeled data. Therefore, the proposed method is unsuitable for obtaining the informativeness of specific data. For this, the method of integrated gradients (IG) was used. The method was developed for image processing neural networks to highlight image areas that are informative for current model outputs [23] and to explain the operation of language ANNs [24]. This method has been adapted for use with the proposed NNC module.

The IG method shows how each input variable for a particular record leads to the output of the ANN model. The mapping is crucial for interpreting ANN decisions, necessary in the context of a computerized medical monitoring system to justify expert choice and decision-making.

The basic idea of the IG method is to integrate the gradients of some loss function concerning the inputs from the base values of the variables (usually zero values) to the values of the specific input variables. The gradients are integrated according to the formula [24, 25]:

$$IG\left(\overline{x}\right)=\left(x_n-x_n'\right)\int_{a=0}^{1}\frac{dF\left(x'+a\left(x-x'\right)\right)}{dx_n}\mathrm{d}a, \quad (11)$$

where $\overline{x}$ is a vector of variable input data, F is a mathematical representation of a neural network, x_n is the n-th variable in the vector of input variables, x_n' is the value of the n-th variable of the base state.

Usually, the values obtained by the IG method are weighting factors for the input data representing the signal flows in the ANN model. Therefore, these values must be normalized using the following equation to represent similar informativeness values:

$$IG_{inform}=\left|IG\left(\overline{x}\right)\right|\Big/\sum_N\left|IG\left(\overline{x}\right)\right|. \quad (12)$$

Considering this and the papers [23–25], the modified algorithm of the IG method can be presented as follows:

1. Determine the base values of the variables; in the cure rent case, these are zeros.

2. Determine the hyperspace lines between the variables' base values and their current ones. Break this line into several points equal to a certain number of steps.

3. Calculate the gradients along the line between the base value and the specific values of the variable at each point.

4. Using Simpson's numerical integration method, integrate the gradients along the line.

5. Normalize the obtained weighting factors according to expression (12).

Modules integration into the stratification subsystem.

It is possible to combine the obtained methods and define data exchange processes in the stratification subsystem. The raw data of the computer medical monitoring system goes to the anomaly detection module, where anomalies are filtered with the specified methods for detecting anomalies. Data filtered from anomalies are sent to the autoencoder module, where the possibility and necessity of data compression are checked. At the autoencoder output, there can be both plain and compressed data. The processed data is used to train the CA module, which allows for determining the number of states and labeling the data according to the states. The labeled data is used to train the ANN model in the NNC module. The SA module uses the trained ANN model and labeled CA data to determine the overall informativeness of the variables. Also, the SA module uses the trained ANN model and flow data to determine the current informativeness, that is, the explanation of the reasons for the state classification by the ANN model.

5. 2. Medical monitoring data application to verify the effectiveness and accuracy of the proposed modules

Dataset description.

Medical data that closely mimicked data in such computerized systems were used to evaluate the proposed stratification subsystem in a computer medical monitoring system. In particular, attention was focused on the diabetes portion of the CDC BRFSS Survey 2021 dataset [26].

The dataset used in this study comes from the Behavioral Risk Factor Surveillance System (BRFSS), which is recognized as the primary health-related telephone survey system in the United States. This extensive system collects state-level data on various aspects of health, including risk behaviors, chronic diseases, and participation in preventive services through annual interviews with more than 400,000 adults. The data set used for this study, specifically the CDC BRFSS 2021 survey, includes responses from 438,693 individuals. In this dataset, 303 variables represent either direct questions asked of participants or derived variables calculated from individual responses.

Diabetes is a long-term health condition that affects the body's ability to convert food into energy. The disease manifests itself in three main types: type 1, type 2, and gestational diabetes. Type 1 diabetes is characterized by an autoimmune reaction that targets the cells in the pancreas responsible for producing insulin, the most important hormone that helps use glucose for energy. Type 2 diabetes, the most common form, occurs when the body shows an abnormal insulin response or does not produce enough insulin. Gestational diabetes, characteristic of pregnancy, occurs during this period and usually disappears after childbirth.

Diabetes data were collected after processing CDC BRFSS 2021 records. The collected dataset comprises 236,378 survey responses, with 21 condition variables most specific to the established disease. The dataset was obtained by removing irrelevant records and combining variables into new ones from the original CDC BRFSS 2021 set [26]. However, each record was labeled with three classes: no diabetes or gestational diabetes, prediabetes, and diabetes status.

Data analysis showed that the data is unbalanced, and about 90 % of the records are individuals who answered no diabetes, but the other variables are not unevenly distributed among the variables. Data was analyzed using appropriate ScikitLearn library functions and distribution visualization using Matplotlib. In addition, correlation analysis between variables was performed using the Pandas library. Any of them is found to be no more than 0.5 or less than –0.3, indicating that reducing the number of variables is difficult. Moreover, there are no variables highly correlated with the target class. In addition, principal component analysis (PCA) was conducted using ScikitLearn tools, taking into account the target class. After analyzing the PCA results (Fig. 3), fifteen variables were required to retain at least 90 % of the information. Fig. 3 is a visualization of the dependence of variance on the number of principal components for the considered data obtained using ScikitLearn and Matplotlib. This result confirms the high diversity of the input variables and the low correlation with the target class. In addition, data were analyzed using two principal components and labeled target clusters. It is challenging to distinguish target classes by simple shapes. All these facts indicate that complex data were presented, which will be a real challenge for the developed system.

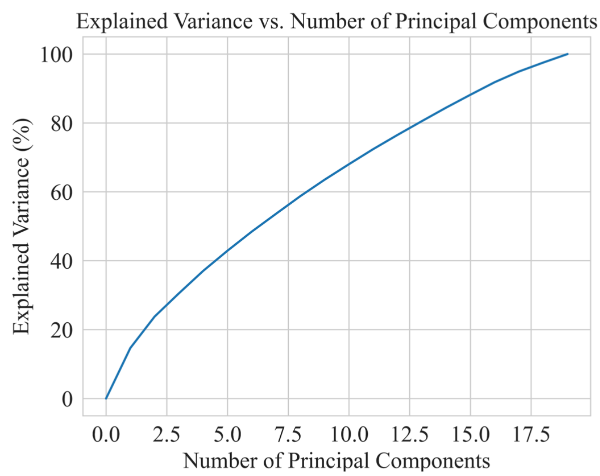


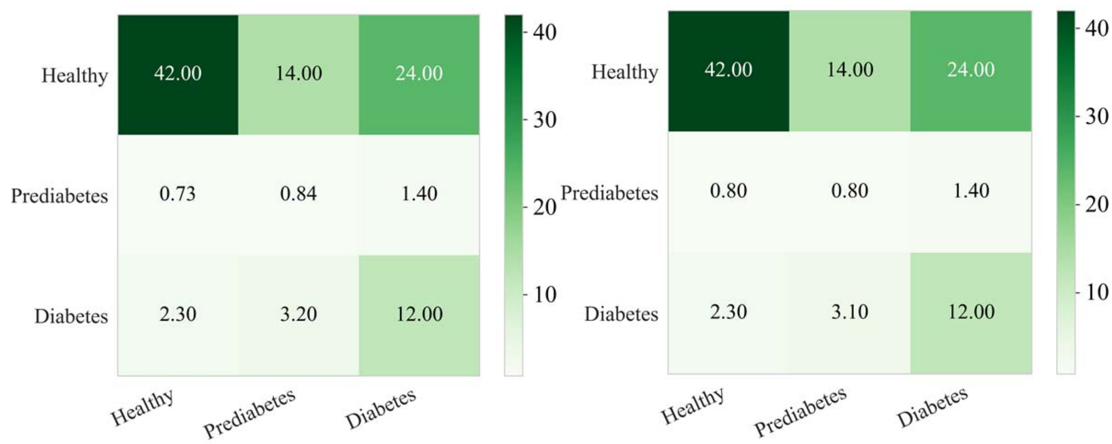
Fig. 3. The explained variance dependence on the number of principal components collected founded on the results of data analysis

Modules testing results.

To analyze the intended functionality of the stratification subsystem, the diabetes dataset was prepared using the specified anomaly detection techniques to reduce the impact of outliers on the training accuracy of the CA module. Due to the low dimensionality of the data, there was no need to use an autoencoder to reduce the dimensionality. With its help, additional data filtering was carried out to detect anomalies that were not there before. The autoencoder model is implemented using the TensorFlow library. The processed data were randomly divided into training and test sets with a standard ratio of 80 % to 20 %. Labeled data was collected after using the CA module, which the system will use in the NNC module for training and further informativeness analysis. The accuracy of the training and test sets was calculated from the confusion matrices (Fig. 4, *a*, *b*): 54.75 % and 54.8 %, respectively. Confusion matrices in Fig. 4 are constructed using the comparison of the results of the CA module and visualized using the Seaborn library.

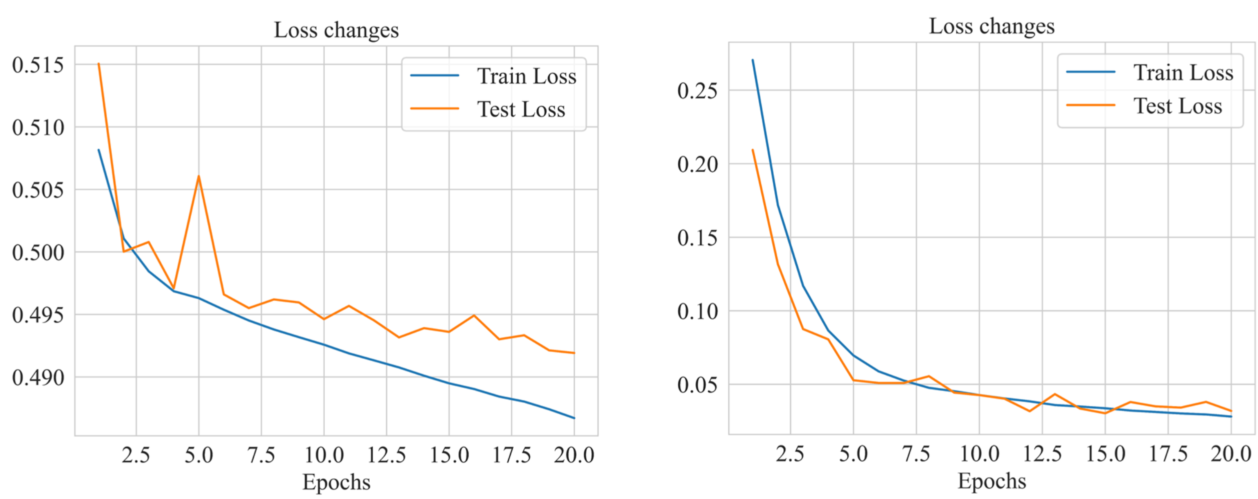
After that, the labeled data from the CA module and the output data with classes for training were sent to the NNC module. The training results of these two neural network models are shown in Fig. 5 and indicate that the NNC module tends to have different loss values on the initially labeled data and generalizes much better to the data labeled by the CA module. The initially labeled data's training/testing accuracy results are 80.8 % and 80.5 %, respectively. Thus, the proposed subsystem builds a model that learns on labeled data and achieves 99.8 % training and 99.7 % test accuracy on labeled data. The loss function values for the test and training sets were used to obtain Fig. 5. These values were collected during the training of two ANN models and visualized by the Matplotlib library.

For further analysis, the SA module with two submodules was considered to obtain overall and current variable informativeness. The result of the overall variables' informativeness is shown in Table 1. Table 1 shows the top 10 variables to achieve a total informativeness of 78.87 %. This method can provide significant information and indicates that the variables HvyAlcoholConsump, CholCheck, Stroke, Age, and HeartDiseaseorAttack have the most significant influence on the decisions of the NNC module. Also, the proposed method of overall informativeness was tested using the feature importance analysis technique with a random forest lateral model [27] and using the permutation importance method [28] on the current ANN (Table 1). The comparison result shows the different nature of the formation of informativeness values. The proposed technique provides a smooth solution from the most to the least informative values. However, the most informative variables collected are valid for test methods.



a *b*

Fig. 4. Confusion matrices indicate the clustering quality: *a* — on training; *b* — on testing sets



a *b*

Fig. 5. Changing the loss function values for training and testing sets during training of the NNC module on the data: *a* — initially labeled; *b* — labeled with the CA module

Table 1

Calculation results of overall informativeness and its cumulative values, as well as Random Forests and Permutation Importance

Variable name	Overall inform.	Cumulative inform.	Inform. Random Forests	Permutation Importance
HvyAlcoholConsump	0.1157	0.1157	0.0711	0.0627
CholCheck	0.1068	0.2226	0.0228	0.1062
Stroke	0.0941	0.3167	0.0244	0.0914
Age	0.0930	0.4098	0.1813	0.0456
HeartDiseaseorAttack	0.0834	0.4932	0.0560	0.1621
AnyHealthcare	0.0839	0.5772	0.0219	0.0727
GenHlth	0.0665	0.6437	0.1032	0.0994
DiffWalk	0.0509	0.6947	0.0653	0.1569
BMI	0.0502	0.7450	0.0984	0.0039
HighChol	0.0437	0.7887	0.0599	0.0109

The use of the IG method to determine informativeness values in the SA module was also considered. Typical results of using this method on some records in the testing set are variables and their informativeness values, such as AnyHealthcare – 17.6 %, PhysActivity – 9.94 %, Stroke – 9.55 %, HeartDiseaseorAttack – 8.91 %, and MentHlth – 8.2 %. So, which variables have the greatest influence on the decisions made by the NNC module can be noted. The results are justified, as some variables are the most informative and in overall informativeness.

Principal component analysis of the diabetes data showed that 15 principal components are required to preserve 80 % of the variance (Fig. 3). In addition, the gradient-based method says that in order to preserve 80 % of the information, it is necessary to preserve ten variables (shown in Table 1, column of cumulative informativeness). This fact indicates that the proposed method can provide acceptable informativeness values for a well-trained NNC module, and this information can be used to determine the most informative variables. Furthermore, the fact that the most informative variables appear to be the most informative in the current informativeness calculated by the IG method indicates that the intended methods can reasonably estimate the informativeness.

6. Discussion of the results of the study of the medical monitoring data usage in the stratification subsystem

As a result of the research, a stratification subsystem was developed with a detailed description of the modules used. The mathematical foundation underlying the proposed methods of anomaly detection, autoencoder, clustering method, and determination of overall and current informativeness is shown. Expressions (10) and (12) show the reduction of informativeness values to a single format. This, in contrast to [23–25], makes it possible to use the IG method for determining current informativeness.

The main task of the proposed stratification subsystem for computer medical monitoring systems is the processing of unlabeled data (in a fully dynamic mode). However, the test results of the SA module, which is responsible for marking the data, showed a 25.7 % lower accuracy compared to the reference results obtained when applying the ANN model trained on the original data (Fig. 5). The reference accuracy results are obtained by the NNC module, which uses an ANN and is essentially a universal approximator. Such discrepancies can be explained by the working principle of the clustering method at the core of the CA module. The proposed clustering model simply cannot capture the complexity inherent in medical monitoring data.

At the same time, the method for determining overall informativeness (Table 1) showed high accuracy compared to analogs. The result can be explained by the assumptions underlying the method of determining overall informativeness, which is correct. It is also worth noting that the relatively small number of variables can introduce inaccuracy in the results of the method analysis and the simplified representation of data classes according to the CA module. The method for determining current informativeness showed results that were not wholly similar to the results of the method of overall informativeness. This can be explained by the slightly different nature of the method's work and the method of ANN analysis.

The advantage of this study is that it tests the stratification subsystem with all modules included on a much larger dataset. This, in contrast to [12], allows testing the multi-agent clustering method on medical monitoring data. Furthermore, unlike [10] – to test the proposed modifications of the ANN on much larger volumes of data. It was also possible to implement the methods of determining informativeness and verify the IG method for the task of medical monitoring. This is in contrast to the works [23–25], where the application of this method was considered for the analysis of convolutional ANNs for image processing. The obtained results make it possible to base the development of a decision support system in computer medical monitoring systems based on the proposed stratification subsystem.

The limitation inherent in the stratification subsystem is the application of the described methods in the corresponding modules. So, instead of the proposed one in the CA module, it is possible to use the DBSCAN or Agglomerative method, which can give different results.

It is also worth noting that an essential drawback of using the system under consideration is the need to involve experts in analyzing the results obtained. This problem is fundamentally impossible to solve due to the extensive nature of data circulating in medical systems and the need to involve the person responsible for decision-making.

The study can be continued by analyzing other methods based on the CA module, which can justify the proposed clustering method's low accuracy. Including the stratification subsystem in the medical decision-making system is also possible. Such a combination will make it possible to check the feasibility of creating a system capable of helping doctors guide the treatment process.

7. Conclusions

1. The study presented modules of the stratification subsystem for complex testing on actual medical data. The subsystem includes anomaly detection and data compression modules using the autoencoder module. Moreover, they can be used if necessary. The stratification subsystem includes

the CA module based on the proposed multi-agent method of fuzzy clustering, the NNC module based on fully connected ANN, and advanced methods of its training. The subsystem also includes the SA module, which has two sub-modules for determining the general and current informativeness of the input variables. The proposed methods in the modules have shown the accuracy of their work in other practical applications. The relative simplicity and cheapness of integration into existing computer systems of medical monitoring distinguish the assembled subsystem. Also, the stratification subsystem in the proposed form has vast possibilities for customization by the tasks to be solved. Among such opportunities, it is possible to highlight the marking of the data flow or the detection of the variables that influence the condition diagnosis by the ANN model in the computer medical monitoring system.

2. The performance of the proposed implementation of the stratification subsystem modules has been evaluated using actual medical data from the CDC BRFSS 2021 survey by selecting diabetes-related data. First, the CA module was tested. It was less accurate than the NNC module trained on the initially labeled data. In addition, the NNC module was trained on the data from the CA module. It showed a high accuracy (99.8 %), indicating that the NNC module can almost perfectly match the data generated by the NNC module. The SA module then calculated the overall and current informativeness of the variables for some randomly selected samples. The SA module results indicate the overall and current informativeness ratio. In addition, the first 10 variables showed similar variance to the PCA estimate.

Conflicts of interest

The authors declare that they have no conflicts of interest in relation to the current study, whether financial, personal, authorship or otherwise, that could affect the study and the results reported in this paper.

Funding

The study was conducted without financial support.

Data availability

The manuscript has associated data in the data repository.

Use of artificial intelligence

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

References

1. Logeshwaran, J., Malik, J. A., Adhikari, N., Joshi, S. S., Bishnoi, P. (2022). IoT-TPMS: An innovation development of triangular patient monitoring system using medical internet of things. International Journal of Health Sciences, 6 (S5), 9070–9084. https://doi.org/10.53730/ijhs.v6ns5.10765
2. Humayun, M., Jhanjhi, N. Z., Almotilag, A., Almufareh, M. F. (2022). Agent-Based Medical Health Monitoring System. Sensors, 22 (8), 2820. https://doi.org/10.3390/s22082820
3. Yu, M., Li, G., Jiang, D., Jiang, G., Tao, B., Chen, D. (2019). Hand medical monitoring system based on machine learning and optimal EMG feature set. Personal and Ubiquitous Computing, 27 (6), 1991–2007. https://doi.org/10.1007/s00779-019-01285-2
4. Singh, V. J., Sharma, P., Mehta, D. A. (2023). Big Data Analytics in Healthcare: Opportunities and Challenges. International Journal of Advanced Research in Science, Communication and Technology, 275–282. https://doi.org/10.48175/ijarsct-9414
5. Yang, J., Soltan, A. A. S., Yang, Y., Clifton, D. A. (2022). Algorithmic Fairness and Bias Mitigation for Clinical Machine Learning: Insights from Rapid COVID-19 Diagnosis by Adversarial Learning. https://doi.org/10.1101/2022.01.13.22268948
6. Ferrario, A., Gloeckler, S., Biller-Andorno, N. (2023). AI knows best? Avoiding the traps of paternalism and other pitfalls of AI-based patient preference prediction. Journal of Medical Ethics, 49 (3), 185–186. https://doi.org/10.1136/jme-2023-108945
7. Goel, K. (2023). Price Discrimination and Product Variety: The Case of Implantable Medical Devices. SSRN Electronic Journal. https://doi.org/10.2139/ssrn.4467475
8. Tamura, M., Nakano, S., Sugahara, T. (2018). Reimbursement pricing for new medical devices in Japan: Is the evaluation of innovation appropriate? The International Journal of Health Planning and Management, 34 (2), 583–593. https://doi.org/10.1002/hpm.2719
9. Strilets, V., Donets, V., Ugryumov, M., Artiuch, S., Zelenskyi, R., Goncharova, T. (2022). Agent-oriented data clustering for medical monitoring. Radioelectronic and Computer Systems, 1, 103–114. https://doi.org/10.32620/reks.2022.1.08
10. Strilets, V., Bakumenko, N., Chernysh, S., Ugryumov, M., Donets, V. (2020). Application of Artificial Neural Networks in the Problems of the Patient's Condition Diagnosis in Medical Monitoring Systems. Advances in Intelligent Systems and Computing, 173–185. https://doi.org/10.1007/978-3-030-37618-5_16
11. Bakumenko, N., Chernysh, S., Bezlyubchenko, A., Goryachaya, V., Donets, V., Strilets, V. et al. (2021). Stratification of Patients in Medical Monitoring Systems based on Machine Learning Methods. https://doi.org/10.13140/RG.2.2.11832.52482
12. Donets, V., Strilets, V., Ugryumov, M., Shevchenko, D., Prokopovych, S., Chagovets, L. (2023). Methodology of the countries' economic development data analysis. System research and information technologies, 4, 21–36. https://doi.org/10.20535/SRIT.2308-8893.2023.4.02
13. Lykhach, O., Ugryumov, M., Shevchenko, D., Shmatkov, S. (2022). Anomaly detection methods in sample datasets when managing processes in systems by the state. Bulletin of V.N. Karazin Kharkiv National University, Series «Mathematical Modeling. Information Technology. Automated Control Systems», 53, 21–40. https://doi.org/10.26565/2304-6201-2022-53-03

14. Shevchenko, D., Ugryumov, M., Artiukh, S. (2023). Monitoring data aggregation of dynamic systems using information technologies. Innovative Technologies and Scientific Solutions for Industries, 1 (23), 123–131. https://doi.org/10.30837/itssi.2023.23.123
15. Prihanditya, H. A., Alamsyah, A. (2020). The Implementation of Z-Score Normalization and Boosting Techniques to Increase Accuracy of C4.5 Algorithm in Diagnosing Chronic Kidney Disease. Journal of Soft Computing Exploration, 1 (1). https://doi.org/10.52465/joscex.v1i1.8
16. Liu, F. T., Ting, K. M., Zhou, Z.-H. (2008). Isolation Forest. 2008 Eighth IEEE International Conference on Data Mining. https://doi.org/10.1109/icdm.2008.17
17. Hinton, G. E., Srivastava, N., Krizhevsky, A., Sutskever, I., Salakhutdinov, R. R. (2012). Improving neural networks by preventing co-adaptation of feature detectors. arXiv. https://doi.org/10.48550/arXiv.1207.0580
18. Ioffe, S., Szegedy, C. (2015). Batch normalization: Accelerating deep network training by reducing internal covariate shift. International conference on machine learning. arXiv. https://doi.org/10.48550/arXiv.1502.03167
19. Pal, N. R., Bezdek, J. C. (1995). On cluster validity for the fuzzy c-means model. IEEE Transactions on Fuzzy Systems, 3 (3), 370–379. https://doi.org/10.1109/91.413225
20. Bridle, J. S. (1990). Probabilistic Interpretation of Feedforward Classification Network Outputs, with Relationships to Statistical Pattern Recognition. Neurocomputing, 227–236. https://doi.org/10.1007/978-3-642-76153-9_28
21. Li, S., Song, Z., Xia, Y., Yu, T., Zhou, T. (2023). The Closeness of In-Context Learning and Weight Shifting for Softmax Regression. arXiv. https://doi.org/10.48550/arXiv.2304.13276
22. Zhang, Z., & Sabuncu, M.R. (2018). Generalized Cross Entropy Loss for Training Deep Neural Networks with Noisy Labels. arXiv. https://doi.org/10.48550/arXiv.1805.07836
23. Qi, Z., Khorram, S., Fuxin, L. (2020). Visualizing Deep Networks by Optimizing with Integrated Gradients. Proceedings of the AAAI Conference on Artificial Intelligence, 34 (07), 11890–11898. https://doi.org/10.1609/aaai.v34i07.6863
24. Enguehard, J. (2023). Sequential Integrated Gradients: a simple but effective method for explaining language models. Findings of the Association for Computational Linguistics: ACL 2023. https://doi.org/10.18653/v1/2023.findings-acl.477
25. Lundstrom, D., Huang, T., Razaviyayn, M. (2022). A Rigorous Study of Integrated Gradients Method and Extensions to Internal Neuron Attributions. arXiv. https://doi.org/10.48550/arXiv.2202.11912
26. Daher, M., Al Rifai, M., Kherallah, R. Y., Rodriguez, F., Mahtta, D., Michos, E. D. et al. (2021). Gender disparities in difficulty accessing healthcare and cost-related medication non-adherence: The CDC behavioral risk factor surveillance system (BRFSS) survey. Preventive Medicine, 153, 106779. https://doi.org/10.1016/j.ypmed.2021.106779
27. Iwendi, C., Bashir, A. K., Peshkar, A., Sujatha, R., Chatterjee, J. M., Pasupuleti, S. et al. (2020). COVID-19 Patient Health Prediction Using Boosted Random Forest Algorithm. Frontiers in Public Health, 8. https://doi.org/10.3389/fpubh.2020.00357
28. Pereira, J. P. B., Stroes, E. S. G., Zwinderman, A. H., Levin, E. (2022). Covered Information Disentanglement: Model Transparency via Unbiased Permutation Importance. Proceedings of the AAAI Conference on Artificial Intelligence, 36 (7), 7984–7992. https://doi.org/10.1609/aaai.v36i7.20769