Available online at www.ijournalse.org

Emerging Science Journal

(ISSN: 2610-9182)

Vol. 8, No. 1, February, 2024

Improving the Quality Indicators of Multilevel Data Sampling Processing Models Based on Unsupervised Clustering

Ilya S. Lebedev [1], Mikhail E. Sukhoparov [2*]

[1] *St. Petersburg Federal Research Centre of the Russian Academy of Sciences, 39, 14th Line V.O., St. Petersburg, 199178, Russian Federation.*

[2] *Russian State Hydrometeorological University (RSHU), 79 Voronezhskaya Ulitsa, St. Petersburg, 192007, Russian Federation.*

Abstract

This paper presents a solution for building and implementing data processing models and experimentally evaluates new possibilities for improving ensemble methods based on multilevel data processing models. This study proposes a model to reduce the cost of retraining models when transforming data properties. The research objective is to improve the quality indicators of machine learning models when solving classification problems. The novelty is a method that uses a multilevel architecture of data processing models to determine the current data properties in segments at different levels and assign algorithms with the best quality indicators. This method differs from the known ones by using several model levels that analyze data properties and assign the best models to individual segments of data and training. The improvement consists of using unsupervised clustering of data samples. The resulting clusters are separate subsamples for assigning the best machine-learning models and algorithms. Experimental values of quality indicators for different classifiers on the whole sample and different segments were obtained. The findings show that unsupervised clustering using multilevel models can significantly improve the quality indicators of "weak" classifiers. The quality indicators of individual classifiers improve when the number of data clusters is increased to a certain threshold. The results obtained are applicable to classification when developing models and machine learning methods. The proposed method improved the classification quality indicators by 2–9% due to segmentation and the assignment of models with the best quality indicators in individual segments.

Keywords:

Regression;

Data Structure;

Prediction;

Simulation.

Article History:

Received:	15	October	2023
Revised:	19	January	2024
Accepted:	27	January	2024
Published:	01	February	2024

1- Introduction

Today, applying artificial intelligence methods to separate tasks improves the achieved qualitative indicators of data processing. Machine learning algorithms enable the identification of the characteristics, statistical properties, and implicit knowledge necessary to achieve a given result by systematically analyzing sufficient relevant data samples. The constantly rising need to improve the qualitative indicators when solving classification, regression, and forecasting problems causes the need to improve data processing methods. Adequate model building depends on the properties of a limited training sample, which should replicate the properties of the general population. A large volume of observation objects is required to achieve high-quality indicators, which is not always possible. Emerging external and internal events in the system can change the properties and characteristics of the analyzed data. Consequently, the problem of model adequacy assessment and the need for retraining arise. However, training many models is unique and requires significant resources, time, research, and experience.

To improve the processing results, models using ensemble methods are applied. They integrate several basic models using simple, weighted, and average voting functions and apply bagging, busting, and stacking methods to form a group

[*] **CONTACT**: msukhoparov.rshu@mail.ru

DOI: http://dx.doi.org/10.28991/ESJ-2024-08-01-025

of algorithms to improve the results [1, 2]. Studies have shown that one of the main advantages of ensembles over individual models is their ability to achieve higher-quality indicators of data processing. By aggregating the results of different underlying algorithms, it becomes possible to level out the erroneous predictions obtained from individual models [3]. However, in practice, there are situations where poorly trained classifiers can degrade the results compared to the application of individual algorithms [4].

Aggregating several basic models provides an opportunity to smooth out the deviations of each algorithm when external influences occur. In transforming the properties of the processed information sequence, the problem of determining noise, outliers, or detecting changes in data properties arises, leading to significant resource consumption for the retraining of classification algorithms [5]. Assigning a separate model to a data segment makes it possible to reduce the labor and resource intensity of these processes and increases the speed of model adaptation to the changed properties of the input sequences of observation objects.

Overtraining can occur during the initial stages of ensemble training [4]. Its manifestation is characteristic of using both complex models and a large number of relatively simple data processing algorithms [6]. Ensemble retraining is a more costly procedure than individual model retraining [5, 7]. Combining multiple models requires solving several problematic issues related to their settings to ensure “distinguishability” in the case of large datasets or limited computational resources [2].

The papers provide various ways of combining ensemble models, but their selection and building are empirical, which does not make it possible to transfer the experience to other subject areas and data structures [2, 3]. Another problem with ensemble approaches is that setting optimal parameters requires considerable knowledge and experience and many analysis procedures [4, 5]. Combining the results of different classification algorithms cannot always reduce the processing error [4, 6]. In addition, there are significant difficulties in forming training subsamples of data aimed at creating *distinguishable* models that give optimal variance and result bias [2]. In any case, the ensemble quality usually depends on the quality of the underlying models. However, the aggregation functions used in ensembles do not always allow us to respond quickly to changes in data properties related to changes in the distribution, frequency of occurrence of observation objects, presence of noise, and outliers, according to Mohammed & Kora (2023) [2] and Akano & James (2022) [5].

This study proposes a three-level model of data processing to solve several identified problematic issues. At the lower level, the model solves information processing tasks. At the middle level, the model assesses the properties of incoming data and makes decisions related to sample segmentation and data separation to improve the quality indicators of processing algorithms. At the upper level, the model monitors quality indicators and performs general management.

This research aims to enhance quality indicators in processing information flows and data samples. The proposed method differs from the known ones by using multilevel models that implement the processes of analyzing data properties, assigning the best models by quality indicators to individual data segments, and training. This study proposes splitting the data sample into separate clusters to improve processing efficiency, which makes it possible to train models on each segment separately and then select and assign the model with the best quality indicators of processing for this cluster from the group of models. The application of unsupervised clustering in generating data segments makes it possible to determine the limit values of processing data quality indicators by changing the number of clusters. This method allows the use of less resource-intensive models to reduce the computational cost of retraining models in the case of changes in data properties.

The rest of this paper is as follows: Section 2 provides an overview of the studies on improving data quality and classification models, highlighting the distinctive features of the proposed solution. Section 3 provides a formalized problem statement and describes the method developed in this study and the application of unsupervised clustering to improve the quality indicators of the data processed on its basis. Section 4 presents the data testing results and discusses the applicability of the method considered in this study based on the conducted experiments. Section 5 provides an interpretation of the obtained results.

2- Literature Review

Improving the quality indicators of classification methods is one of the essential problems in machine learning.

To date, one of the main directions for improving the quality of processed data is to combine models into various ensembles. Interest in such methods does not fade despite the prevalence of the neural network approach [8, 9] because the need arises to implement hybrid models that combine deep learning methods with classical classification algorithms to improve quality indicators in data processing. Such symbiosis for particular tasks makes it possible to significantly improve quality indicators [10, 11]. However, such models have complexities in interpreting results, are prone to saturation of the neural network, and are difficult for retraining processes when the properties of the processed data change.

Ensemble data processing techniques have been improving since their introduction. Studies on the aggregation of data processing models constantly consider *sampling* ensembles using various weighted sampling voting functions, cascades of simple algorithms, and deep neural networks, according to Huang et al. (2023) [12] and Brown et al. (2023) [13]. The task of such functions is to remove the dependence of the underlying algorithms on each other by averaging the results. Each function has its advantages and disadvantages. Implementing weighted voting requires constant updating of the weight coefficients of the underlying algorithms of simple voting, which are not sensitive to changes in the data properties; the applied cascades are highly dependent on the initial settings and complicated to reconfigure.

Using algorithm ensembles improves quality indicators of processing in many data mining tasks by evaluating the results of different processing methods. It creates a more accurate model that aggregates the output results, helps improve the prediction result, and reduces the model's dependence on a particular dataset [14]. All these methods improve some qualitative indicators to a greater or lesser extent, but their main disadvantages are training complexity, resource intensity, and increased algorithm running time.

Another direction for improving the indicators of data processing quality is sampling. The modern approach to machine learning methods is defined by the paradigm of “model plus data,” in which both model and data have the same significance. Data-splitting techniques improve the quality indicators of processing and optimize learning processes. Sampling separation for processing models is an auxiliary element of the ensemble methods, which involves grouping the data first and then independently training algorithms on subsamples. However, the issue of forming segments optimized to improve the performance of models trained on them is not usually considered [15, 16].

Many datasets have complex base structures. When applying, for example, different classification algorithms to the whole sample, situations may arise that affect the achievement of the given quality indicators. Linear classifiers lose completeness and accuracy in the case of nonlinear data distribution. Nonlinear classifiers and neural networks require large training datasets and resources to achieve quality results. However, if segments are detected in the data, using such information in many cases will improve the processing quality for both linear and nonlinear classifiers [17, 18].

Deep analysis for sample separation often reveals relatively homogeneous characteristics in segments, which provides advantages for building processing models [19, 20]. Investigated various aspects of vertically separated data, proposed techniques, basic algorithms, and combination strategies aimed at selecting observation objects [21, 22]. This makes it possible to obtain the main characteristics of sequences and samples and exclude values that lead to distortion of properties [23]. Djouzi et al. (2022) [24] determined the principles of data segmentation, investigated the effect of clustering on the quality of SVM classifier predictions, and concluded that there is a relationship between the number of clusters and classification quality.

However, using the described methods incurs significant costs for analyzing the processed data [25, 26]. In their application, situations may arise when they deteriorate instead of improving quality indicators [27, 28]. Reduction of computational costs for analysis and determination of internal data structures is possible based on unsupervised clustering, which defines clusters by grouping objects based on their similarity without prior knowledge of the number of clusters [29, 30]. Such approaches lose qualitative indicators to some processing models but, compared with other algorithms, do not require significant costs for deep data analysis. Clustering makes it possible to identify groups of similar objects, consider outliers, and identify atypical objects. Belonging to a cluster and analyzing the properties of objects belonging to it (distributions, value ranges, incidence of classes) provide additional information for the machine learning method.

The most popular modern machine learning methods, such as ANFIS, RBF, ANN, NB, LD, SVM, DT, and K-nearest neighbors (KNN), as well as their ensembles, are often used for data processing in various subject areas. The analysis of the outlined sources shows some shortcomings of the state-of-the-art models. ANFIS suffers from limitations related to dimensionality and computational costs [31]. RBFs have poor extrapolation properties [19]. ANN has non-guaranteed convergence and significant training costs [31, 32]. NB uses a not always correct assumption of feature independence [18]. LD is sensitive to data distribution [20]. SVM performs poorly in the presence of outliers and noise [15]. DT is susceptible to uncontrolled growth in the presence of some variants [34]. RNN has temporal and spatial complexity [24]. Incorporating additional information about data clusters can improve quality indicators of processing in some cases. Nevertheless, choosing the best method is difficult because the space of a single or mere hypothesis of learning algorithms shows difficulties in satisfying all possible scenarios as the data used changes, according to Wei et al. (2022) [34].

The overwhelming majority of methods have disadvantages related to computational complexity, processing of noisy information, sensitivity to dimensionality, distributions, and statistical properties of processed data. In this regard, it is possible to use multilevel models that can analyze and divide the sample into separate segments according to their properties and assign the most effective processing model to each segment. Thus, there is a need to develop new and adapt existing strategies that enable accurate and robust learning within the separation of functions and samples. Data becomes an essential component, affecting the quality of models within the paradigm under consideration. Data and their properties are fundamental to model selection and ensemble building.

This paper proposes a possible solution to improve quality indicators, which considers the application of unsupervised clustering of the data sample in multilevel models for solving classification problems. It investigates the possibility of assigning a model to a cluster that shows better results on the objects belonging to it than other algorithms. This study also determines the limits of the values of quality indicators based on the number of clusters into which the sample is divided.

3- Proposed Model

3-1- Basic Notation

Currently, most machine learning methods are highly dependent on the properties of the processed data. Each model is optimized to achieve high-quality indicators for the training samples, which consist of a predetermined set of observation objects. In the case of shifts in the ranges of predictors and target variables, changes in distributions in information sequences processing quality indicators may decrease, which leads to the necessity of reapplication of training methods. Model retraining processes require a considerable time and resources. When phenomena such as concept drift occur, the processing model may lose its relevance by the time of retraining.

Therefore, this study considers a solution aimed at reducing the resource intensity of learning processes based on the segmentation of samples, on which, depending on the data properties, the model selection with the best quality indicators occurs, implemented by generating data processing models consisting of several levels. Each of them solves predetermined tasks: analysis and segmentation of incoming information flows, training of predetermined models, and assignment of the model with the best qualitative indicators for the current segment. Figure 1 shows the three-level data processing model.

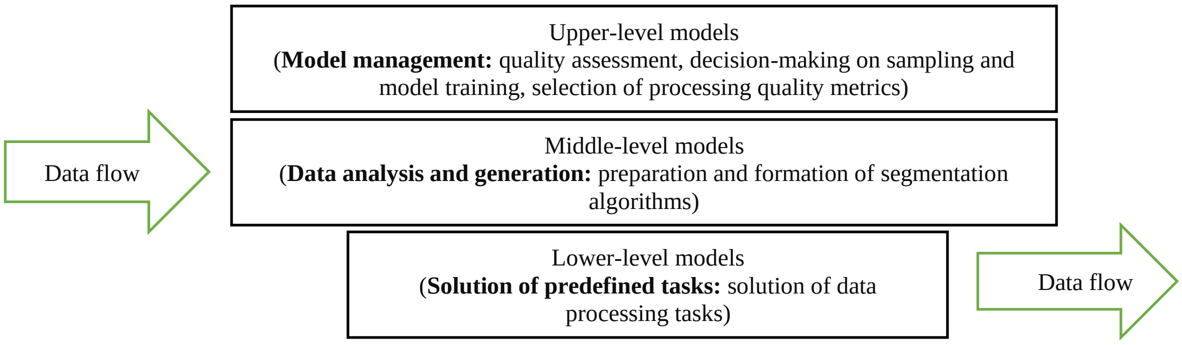

Upper-level models
(**Model management:** quality assessment, decision-making on sampling and model training, selection of processing quality metrics)
Data flow
Middle-level models
(**Data analysis and generation:** preparation and formation of segmentation algorithms)
Lower-level models
(**Solution of predefined tasks:** solution of data processing tasks)
Data flow

Figure 1. A three-level data processing model

At the lower level, the model performs tasks such as forecasting or classification. At the middle level, the model performs tasks arising from changing properties of incoming data associated with finding cluster structures, time series change points, and concept drift detection. At the upper level, the model monitors quality indicators and performs general management and model assignments.

In many systems that process information flows, due to external and internal influences, data properties may change or return to the initial state after particular time intervals. Therefore, it is necessary to monitor the incoming sequences and segment them by considering the influencing factors. The aim of dividing the sample is to obtain data subgroups to build relatively simple separating surfaces between their classes, thus minimizing errors. It is desirable to minimize computational complexity and resource intensity.

3-2- Formal Description of the Proposed Method

Models of different levels interact on the basis of the analysis of the current properties of the processed data. The initial stage of multilevel structure formation implies the presence of predetermined sets of data processing models $a_1, \dots, a_r$ and data sample separation functions $\mu_1, \dots, \mu_m$. For data analysis and initial adjustment of the models, a data sample X reflecting the properties of the general population is required.

Different separation functions form segments in the sample depending on the type of input data. These can be, for example, methods of searching for change points for time sequences, determining concept drift, and sample clustering. The sets of observation objects in the obtained segments may have different properties. Any data processing models $a_1, \dots, a_r$ in different segments will show different quality indicators. Therefore, it is necessary to evaluate the quality indicators of the segment data processing models obtained using different functions $\mu_1, \dots, \mu_m$ of sample separation. Each function $\mu_l: X \rightarrow \{1_{\mu_l}, \dots, M_{\mu_l}\}$ splits the sample X into a different number of segments M_{μ_l}.

By defining separation functions $\{\mu_1,\dots,\mu_m\}\in M$ as clustering methods, for each μ_l it is possible to obtain different separations of the data sample $\{X_{1_{\mu_l}}^{\mu_l},\dots,X_{M_{\mu_l}}^{\mu_l}\}\in X$, differing in the number of clusters and the composition of observation objects. Setting different parameters for the same algorithm, for example, applying various distance measures, gives groups of clusters with significant differences in data shapes and properties. For a chosen separation function μ_l on each segment $X_i^{\mu_l}\in X$ obtained using its application, it is possible to determine processing models $a_j^{\mu_l}(x)$ that achieve better results.

$$a_j^{\mu_l}(x) = \underset{a_j^{\mu_l}\in A,\mu_l\in\mu}{argmax}\, Q(a_j^{\mu_l}(x), X_i^{\mu_l}) \tag{1}$$

Equation 1 allows us to determine the processing model assigned to a segment when separating the data sample. Due to possible limitations in the use of computational resources, it is necessary to initially limit the number of data processing models in a given multilevel model and to determine the limit of the number of possible clusters for clustering methods. For the chosen function μ_l, the separation of the data sample is multiple and depends on the given limit. By applying the function μ_l, different separations of the data sample containing from 2 to M limits of clusters appear. The values of the achieved quality indicators of all processing models in each cluster were calculated for the obtained separations. The evaluation of the achieved indicators uses the averaged quality functional for each clustering method when processing by different models:

$$Q_{\mu_l} = \frac{1}{M_{\mu_l}}\sum_{k=1}^{M_{\mu_l}}\left[max \sum_{j=1}^{n} Q(a_j^{\mu_l}(x), X_k^{\mu_l})\right] \tag{2}$$

The following expression determines the choice of clustering method based on Equation 2:

$$\mu = \underset{\mu_l}{argmax}(Q_{\mu_1,\dots,}\, Q_{\mu_l}) \tag{3}$$

Thus, the initial sample is separated into several parts. Then, pre-selected models $a_1, a_2, \dots, a_r$ are trained on clusters, determining the values of the quality functional $Q(a_j^{\mu_l}(x), X_k^{\mu_l})$ achieved by each model. Based on their values, the models are ranked, determining for each segment those models from the set $\{a_1, a_2, \dots, a_r\}\in A$ that have the highest quality indicators.

The observation objects of the information flow coming to the input are processed, determining their belonging to the cluster selected in the training sample and assigning the processing model with the best values of the quality functional on a similar segment of the training sample. The obtained results were analyzed. If the errors increase above a predetermined threshold, a decision is made to generate data for algorithm refinement, which is subsequently added later to the training sample.

3-3- Method Implementation

The proposed method involves several operations necessary to customize the models and perform predefined tasks. However, some limitations need to be considered during implementation. Figure 2 shows the sequence of data processing. This method requires a training dataset that repeats the properties of the analyzed dataset. Initially, it is assumed that there are sets of predetermined data processing algorithms and clustering methods. Data processing involves several steps:

1. Generating a training dataset *X* containing training examples;
2. Selecting several clustering methods $\mu_1,\dots,\mu_m$for sample *X*. Each clustering method processes the training dataset. The number of clusters k is a varying parameter $k = 1, \dots, M$. For each clustering method, several datasets appear, differing in the number of clusters;
3. Determining the number of considered datasets using the number of clustering methods and the maximum number of clusters;
4. Training data processing models$a_1,\dots,a_r$ on all clusters of the obtained datasets$X_i^{\mu_l}\in X$, calculating the quality indicators of all models for each;
5. Determining the best method of data clustering μ_i according to a given quality indicator of the data processing models and the resulting number of clusters;
6. Creating for the clustering method μ_i, a classification model that includes a group of algorithms $a_1^{\mu_l}(x, X_{1_{\mu_l}}^{\mu_l}), \dots, a_M^{\mu_l}(x, X_{M_{\mu_l}}^{\mu_l})$, ..., performing processing on each segment.

Thus, clustering sample *X* sets the maximum number of clusters based on the availability of computing resources. Then, all models are trained on each cluster to solve the required information processing tasks, generate the values of qualitative indicators for each model, and assign the best model to each cluster.

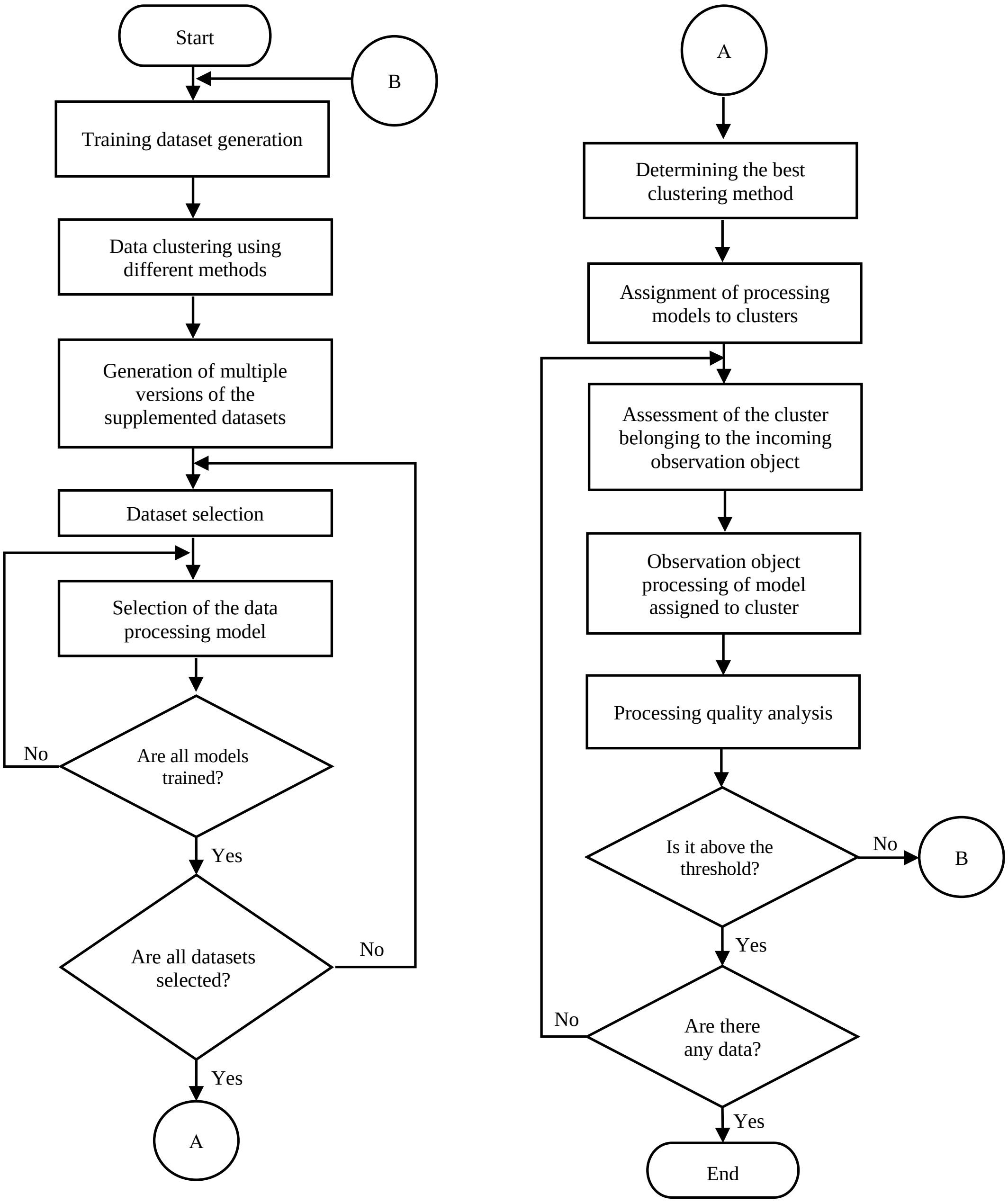

Start
B
Training dataset generation
Data clustering using different methods
Generation of multiple versions of the supplemented datasets
Dataset selection
Selection of the data processing model
No
Are all models trained?
Yes
Are all datasets selected?
No
Yes
A
A
Determining the best clustering method
Assignment of processing models to clusters
Assessment of the cluster belonging to the incoming observation object
Observation object processing of model assigned to cluster
Processing quality analysis
Is it above the threshold?
No
B
Yes
No
Are there any data?
Yes
End

Figure 2. Flowchart of the data processing

Figure 3 shows a general view of the processing algorithm.

Algorithm 1: Processing algorithm

```
Input: flow sample x, dataset $X$, methods $\{a_1,\ldots,a_r\}\in A$, split methods$\{\mu_1,\ldots,\mu_m\}\in M$
Output: quality indicator $q$, method for flow sample $a(x)$ split methods $\mu$
begin
for $l = 1$ to $m$
{
for$k=1$ to $M$
 {
$\mu_l: X \rightarrow \{X_1^{\mu_l},\ldots,X_k^{\mu_l}\}$       //Generation of clusters $X_1^{\mu_l},\ldots,X_k^{\mu_l}$by $\mu_l$ method
for $i=1$ to $k$//cluster searching
 {
for $j=1$ to$r$//model searching
 {
$q_{i,j}^{\mu_l} = Q(a_j(x), X_i^{\mu_l})$ //determining the value of the quality indicator for all models in the cluster$X_i^{\mu_l}$
$a_{i,j}^{\mu_l}(x) \leftarrow q_{i,j}^{\mu_l}$// identifying the model that has achieved the quality indicator
}
$q^{\mu_l,i} = \max(q_{i,j}^{\mu_l})$// selection of the best quality indicator value in the cluster $X_i^{\mu_l}$ for all models $a_j \in A$
$a^{\mu_l,i}(x) \leftarrow q^{\mu_l,i}$// identifying the model that has achieved the value of the quality indicator
 }
$q^{\mu_l,k} = \frac{1}{k}\sum_{i=1}^{k} q^{\mu_l,i}$// calculation of the average quality indicator for $k$ clusters obtained from method $\mu_l$
$a^{\mu_l,k}(x) \leftarrow q^{\mu_l,k}$ // determining the aggregate model (consisting of models assigned to clusters),
// which has reached the value of the quality indicator
 }
$q^{\mu_l} = \max (q^{\mu_l,k})$// selecting the best quality indicator value for all models $a_j \in A$
 // using $\mu_l$ method at separations from 1 to M clusters
$a^{\mu_l}(x) \leftarrow q^{\mu_l}$// determining the aggregate model that has reached the value of the quality indicator
}
$q = \max (q^{\mu_l})$// selecting the best quality indicator for all models
     // when clustering using μ methods with different numbers of clusters
$a(x) \leftarrow q$//determining the aggregate model that has reached the value of the quality indicator
$\mu = \max_{\mu}(q^{\mu_l})$// identifying the clustering method that achieved the value of the quality indicator
end
```

Figure 3. General view of the processing algorithm

Later, when a new observation object comes for processing, its cluster membership is determined. The incoming observation object, supplemented with cluster membership information, is passed to the data processing model assigned to process this cluster. The model makes a prediction. All actions of the machine learning algorithms occur with separate data regions selected in the preliminary stage. However, with this approach, one of the main problematic issues is separating the sample into clusters in which the objects have particular properties. An essential disadvantage of clustering methods is the need to select the number of clusters. Simultaneously, homogeneity problems appear in different situations, where a cluster can have another range of values, distributions, forms of representation, and others. There are many ways to determine the number of separations; however, under the conditions of possible changes in data properties, their application does not always have a proper effect on quality indicators of processing. In this regard, we consider using unsupervised clustering with a previously unknown number of clusters.

4- Evaluation

4-1- Experimental Setting

When experimenting, clustering was chosen as the method of data separation [35, 36]. It separates observation objects of the sample with high similarity into one cluster and those with low similarity into different clusters, according to Tong et al. (2023) [37]. There are many clustering methods [38, 39]; however, the algorithms considered in the experiment are hard clustering algorithms. Observation objects in the set belong to only one cluster. Figure 4 shows the framework of the experimental implementation. The preparatory stage involves training set formation. Then, metrics that determine quality indicators of processing are selected. The sets of algorithms for separating the sample into clusters and the data processing models under study are defined.

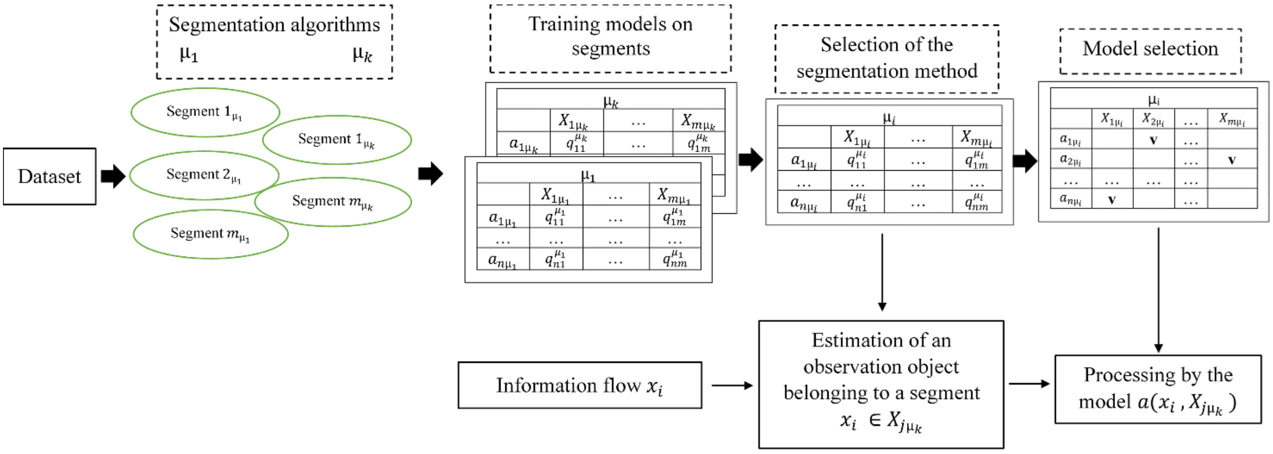

Segmentation algorithms
μ_1
μ_k
Dataset
Segment 1_{μ_1}
Segment 1_{μ_k}
Segment 2_{μ_1}
Segment m_{μ_k}
Segment m_{μ_1}
Training models on segments
μ_k
$X_{1\mu_k}$ … $X_{m\mu_k}$
$a_{1\mu_k}$ $q_{11}^{\mu_k}$ … $q_{1m}^{\mu_k}$
μ_1
$X_{1\mu_1}$ … $X_{m\mu_1}$
$a_{1\mu_1}$ $q_{11}^{\mu_1}$ … $q_{1m}^{\mu_1}$
$a_{n\mu_1}$ $q_{n1}^{\mu_1}$ … $q_{nm}^{\mu_1}$
Selection of the segmentation method
μ_i
$X_{1\mu_i}$ … $X_{m\mu_i}$
$a_{1\mu_i}$ $q_{11}^{\mu_i}$ … $q_{1m}^{\mu_i}$
$a_{n\mu_i}$ $q_{n1}^{\mu_i}$ … $q_{nm}^{\mu_i}$
Model selection
μ_i
$X_{1\mu_i}$ $X_{2\mu_i}$ … $X_{m\mu_i}$
$a_{1\mu_i}$ v
$a_{2\mu_i}$ v
$a_{n\mu_i}$ v
Information flow x_i
Estimation of an observation object belonging to a segment $x_i \in X_{j\mu_k}$
Processing by the model $a(x_i, X_{j\mu_k})$

Figure 4. Implementation framework for the data processing experiment

Clustering algorithms $\mu_1,\ldots,\mu_m$, having different parameters that determine the number of clusters and types of distance functions between them, process the training sample. For each clustering algorithm, sets of segments$X_{1_{\mu_l}}^{\mu_l},\ldots,X_{M_{\mu_l}}^{\mu_l}$ are generated and considered separately. Each segment of the set feeds into the inputs of the models $a_1,\ldots,a_r$, and the models are trained on them to determine the quality measures they achieve. Then, for each separation method, the processing models analyze the achieved processing quality indicators to select the best method of sample separation and generate a processing model. In sample segmentation, the effect of the number of clusters on the quality indicators of the model is considered.

4-2- Model Data Processing

In the experiment, K-means was used as the basic clustering algorithm. During the training processes, the processed sequences were separated into clusters, considering different distance measures (Euclidean measure, "city blocks" distance measure, correlation coefficient-based measure, "cosine measure"), which influenced the shapes of the clusters. In each cluster, the value of its center was calculated. Data processing models were trained on the selected clusters to determine the optimal values of the quality functional. Subsequently, the remaining part of the records simulated the information flow. According to the incoming values, cluster membership determination uses the distance function between objects. The share of correct answers (accuracy) served as the analyzed indicator.

$$Accuracy = \frac{correctclassification}{allclassification} \times 100\% \tag{4}$$

All samples were processed entirely by clusters using different algorithms. In the considered example, each model (naive Bayesian classifier NB, linear discriminant LD, support vector method SVM, decision trees DT, K-nearest neighbors KNN, ensemble model of all classifiers ENS) was trained on the sequence as a whole and on the data subsets obtained because of clustering using four distance measures. Figure 5 shows an example of the model data under consideration.

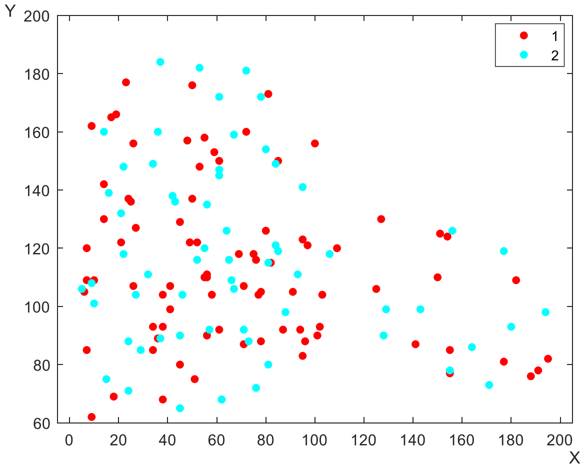
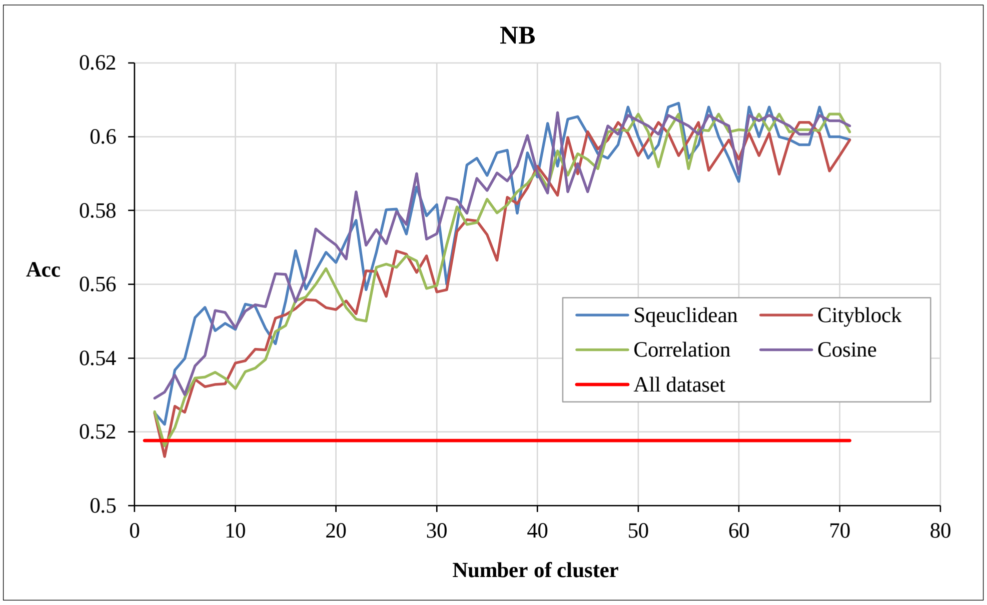
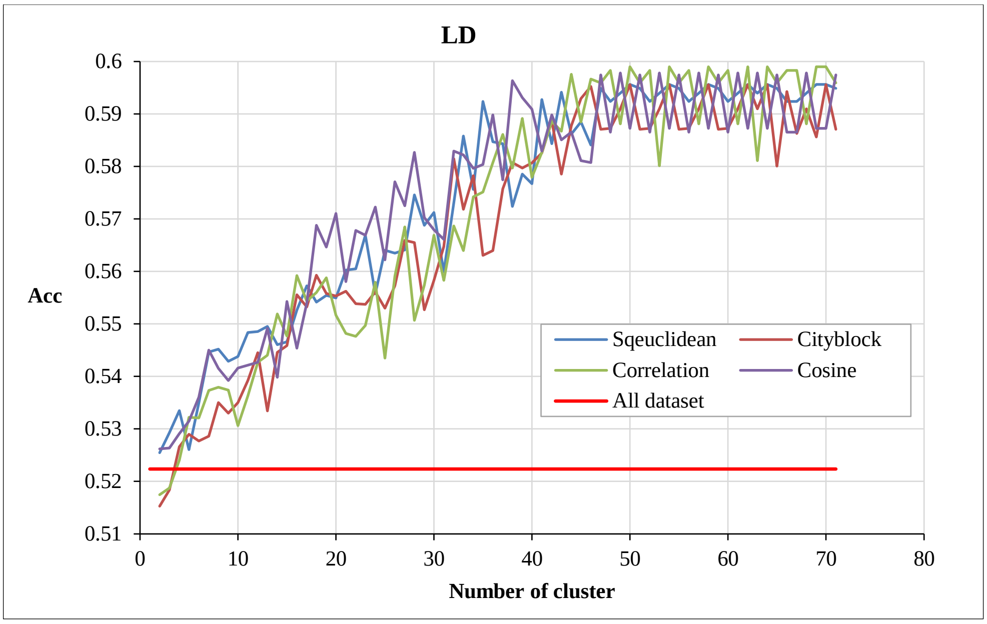
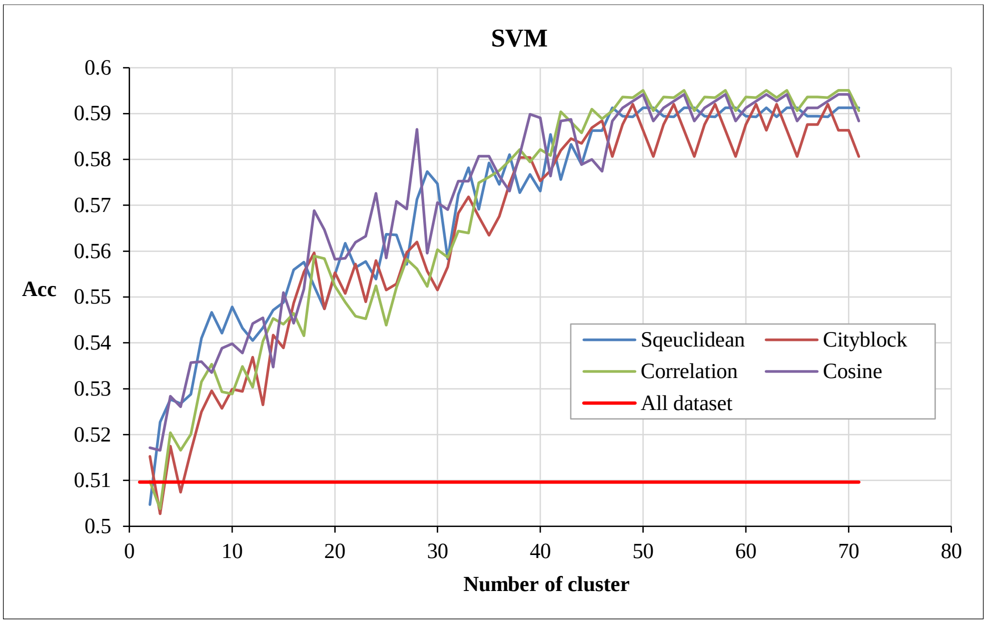


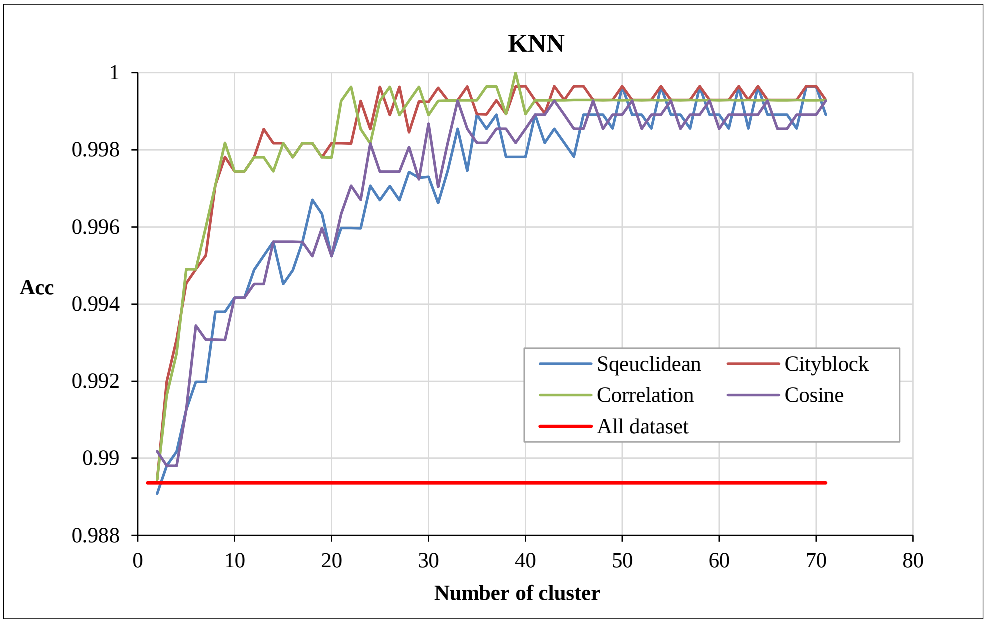
Figure 5. Model data series of X,Y values for two classes

To implement the assumed model, the first step is to divide the set into separate clusters. The obtained clusters should satisfy the functional quality criteria (3). In the experiment, the number of segments was determined by the achievable processing quality of the data processing models. Figure 6 shows the accuracy (Acc) indicators when increasing the amount of sample separating clusters from 2 to 70 for each classifier (accuracy values of models when processing the whole sample without clustering, average accuracy values of models when forming clusters using Euclidean distance measure, using city blocks distance measure, using correlation measure, using cosine measure).




LD
Acc
0.6
0.59
0.58
0.57
0.56
0.55
0.54
0.53
0.52
0.51
0
10
20
30
40
50
60
70
80
Number of cluster
Sqeuclidean
Cityblock
Correlation
Cosine
All dataset


SVM
Acc
0.6
0.59
0.58
0.57
0.56
0.55
0.54
0.53
0.52
0.51
0.5
0
10
20
30
40
50
60
70
80
Number of cluster
Sqeuclidean
Cityblock
Correlation
Cosine
All dataset


KNN
Acc
1
0.998
0.996
0.994
0.992
0.99
0.988
0
10
20
30
40
50
60
70
80
Number of cluster
Sqeuclidean
Cityblock
Correlation
Cosine
All dataset

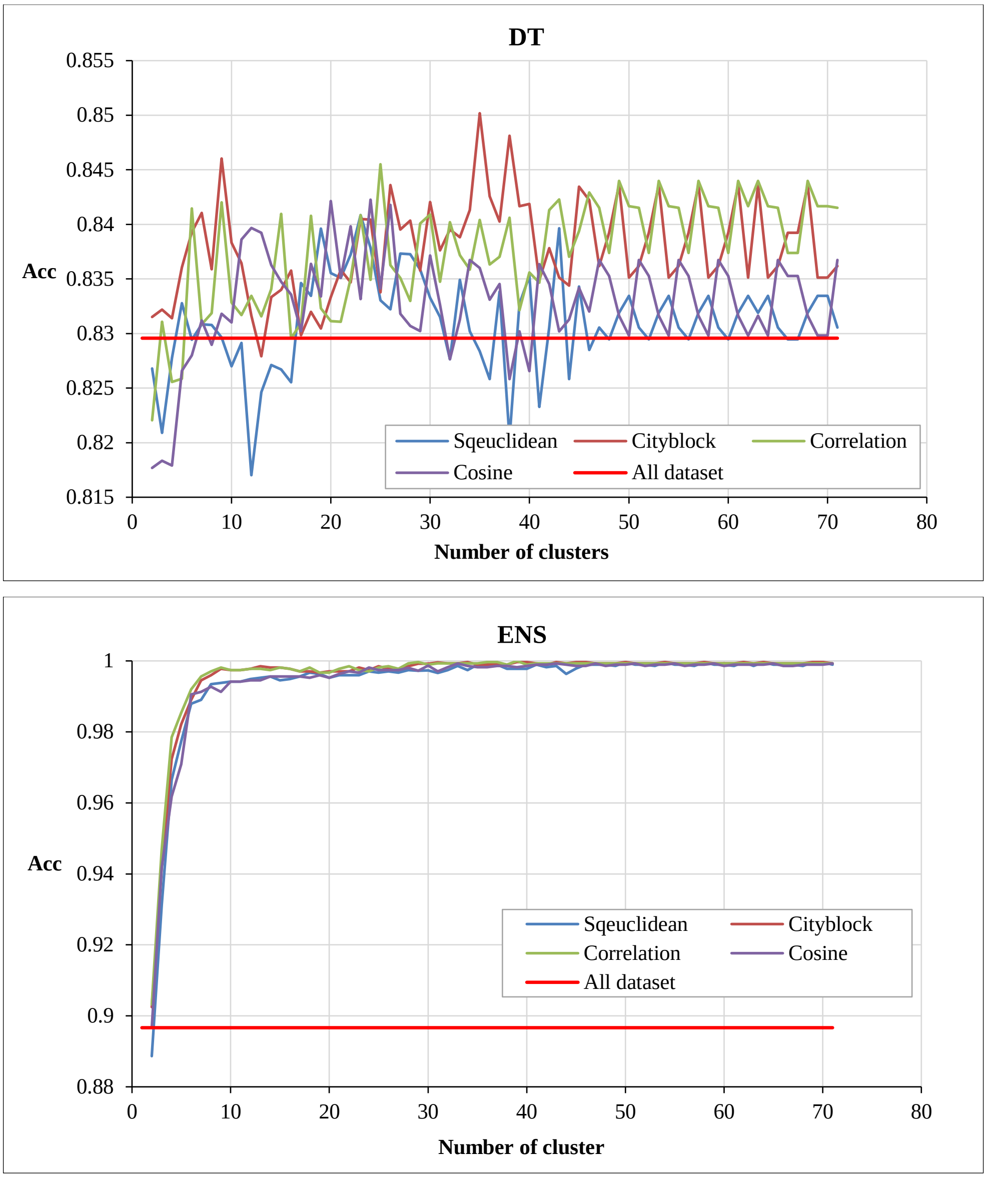


Figure 6. Results of the data processing models

The graphs in Figure 6 show that the number of clusters influences the achieved values of the accuracy indicator. The example of the share of correct answers shows that, up to a certain point, the more clusters there are, the higher the values of quality indicators. After that, the quality indicators of the model reach a plateau, and a further increase in the number of clusters does not lead to a significant improvement in the classification quality. In some cases, separation into clusters leads to the deterioration of processing model performance because, in unsupervised clustering, each cluster may have a different data distribution, and some algorithms, such as decision trees (DT), do not receive enough information to determine the data distribution.

The shape of clusters, defined by a measure of the distance between points, can affect the performance of processing models. The emergence of *complex shapes* can complicate the construction of an effective separating surface. In addition, when the number of clusters is unlimited, *microclusters* containing only observation objects of the same class appear. When applying the decisive rule for assigning a new object based on its proximity to the center of the nearest cluster, a *microcluster* containing only objects of the same class will be assigned the current class. However, based on the training sample size, it is not always correct.

The proposed solution makes it possible to use clusters as additional information for model training. However, clusters are often heterogeneous, and some algorithms may perform better on some segments, whereas other algorithms perform better on others. By dividing the models of the considered example into two groups ("strong" classification algorithms, where the accuracy value is close to 1, and "weak" algorithms, which produce results between 0.5-0.7), it is possible to see the improvement of processing data quality indicators by segments compared to the whole sample. Figures 7 and 8 show the accuracy (Acc) values for different groups ("strong" and "weak") of classifiers when separating the sample into 14 segments. Each algorithm for processing data falling into one of the numbered clusters has a value for the share of correct answers.

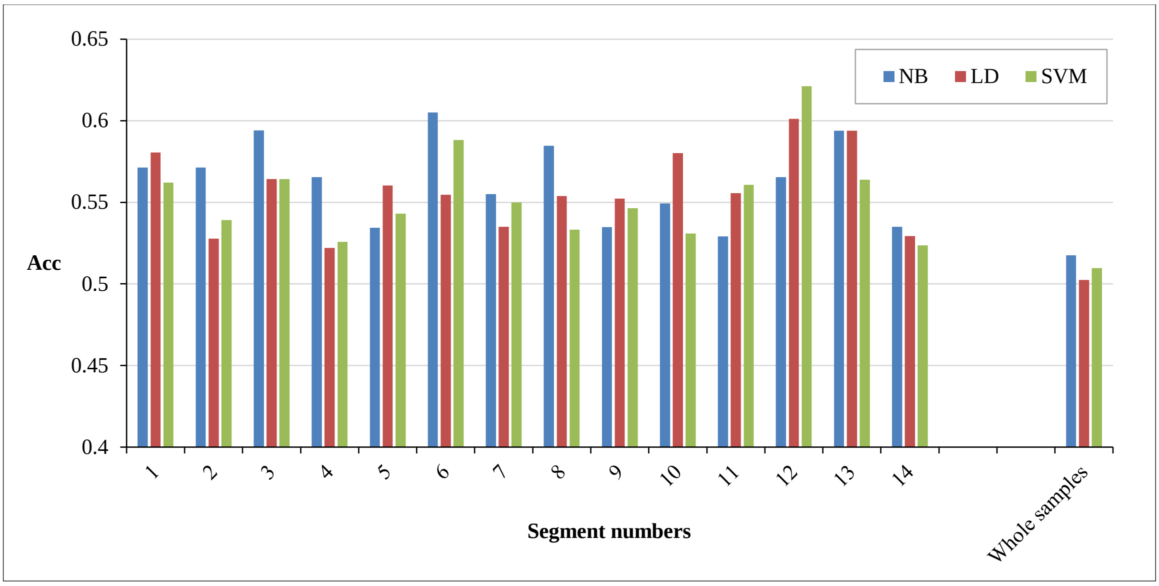


Figure 7. **Performance of “strong” data processing models on different clusters**

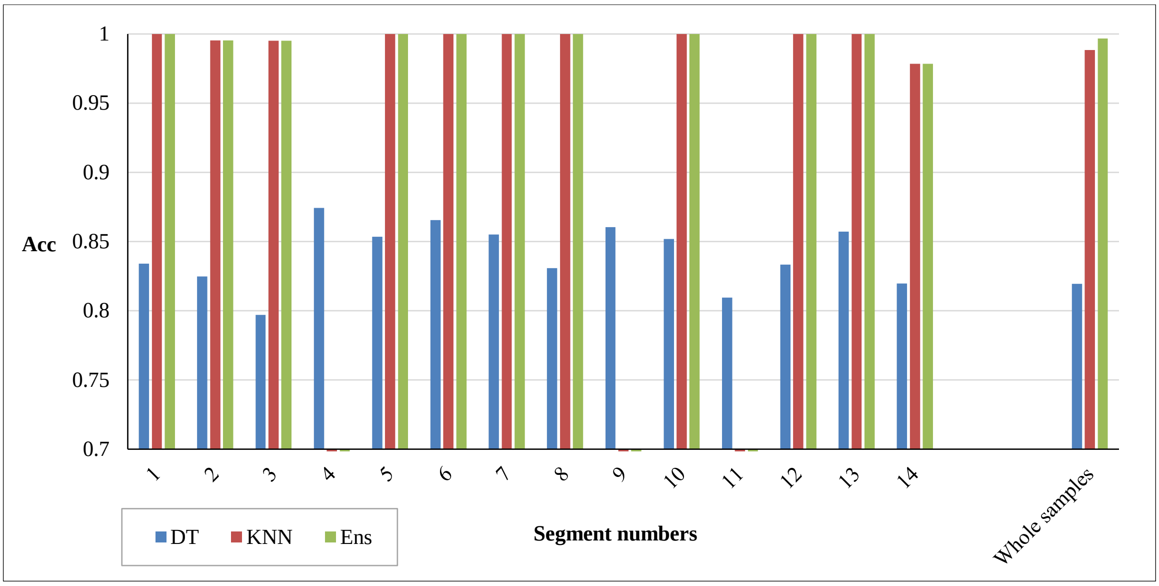


Figure 8. **Performance of “weak” data processing models on different clusters**

The histograms in Figures 7 and 8 show that for this sample, clustering allows a gain for each model when the classifier analyzes a segment individually compared to the whole sample. Figure 8 considers three “weak” processing models (NB, linear discriminant LD, and linear SVM). In the case of separating the sample into 14 clusters, it is possible to assign a better performing algorithm to each cluster, which improves the results compared with processing the whole sample.

Increasing the number of sample clusters leads the data processing models to a definite limit of quality indicators, after which such an increase does not have a proper effect. Reducing the cluster "size" can lead to the fact that, for example, in classification tasks, it is possible to build a simplified separating surface for data separation. On the one hand, this allows for improving the quality of individual algorithms and reducing computational complexity, but on the other hand, it causes problems of representativeness, homogeneity, and sample adequacy. Each segment can affect the algorithm in different ways, either improving or worsening its performance. The shape of the clusters resulting from the application of different distances can affect the quality indicators of the data processing models. In the case of applying one distance measure in clustering, a particular set of algorithms that efficiently process clusters is possible; however, for another one, the composition of algorithms may change. By assigning a processing algorithm with better quality indicators to each segment, it is possible to obtain a gain in the indicator compared with processing the entire sample using one model.

4-3- *Algorithm Evaluation*

The next stage of the experiment considered several available datasets and results obtained by other researchers [40–47], adding the following models: adaptive neuro-fuzzy inference system (ANFIS), radial basis function (RBF) neural network, and neural network (ANN). Table 1 shows the proportion of correct responses in the data processing models with and without segmentation.

Table 1. Characteristics of the datasets and results of the accuracy of the processing models

Database/Dataset	Classification model	All dataset	Proposed solution (segmentation)
PIMA	CNN	73.54%	**80.1%**
PIMA	DT	78.01%	**82.03%**
PIMA	LD	73.11%	**80.1%**
PIMA	LR	75.72%	**77.31%**
PIMA	NB	73.12%	**79.18%**
PIMA	NN	72.1%	**72.6%**
PIMA	RF	73.51%	**74.01%**
PIMA	SVM	92.6%	**96.8%**
PCU	ANN	82.96%	**85.34%**
PCU	DT	84.09%	**84.11%**
PCU	LR	79.9%	**86.01%**
PCU	NB	78.91%	**84.65%**
LCLDA	KNN	83.14%	**87.21%**
Hepatitis	RBF	75.31%	**80.12%**
Compustat	ANFIS	90.22%	**94.31%**

Table 2 shows the results of the proportion of correct responses of the data processing models on selected segments and on the whole sample (without segmentation) for the ensemble models of simple voting and assigning models.

Table 2. Characteristics of the datasets and results of accuracy of the processing ensemble models

Database/Dataset	Classification ensemble model	Result simple voting model	Proposed solution (segmentation) +purpose model
PIMA	NB+DT	78.9%	**84.23%**
PIMA	NB+LR	75.81%	**79.31%**
PCU	NB+LR+DT	82.1%	**86.9%**

Table 3 shows the results of the proportion of correct responses of the data processing models on selected segments and the entire sample when the models reach the limit of the quality indicator of accuracy (Equation 4).

Table 3. Comparison of the obtained quality indicator of accuracy with the results of other researchers

Database/Dataset	Classification model	Result	Proposed solution (segmentation)
PIMA	NB (Iyer & Sumbaly, 2015) [41]	76.95%	**79.18%**
PIMA	RF (Zou et al., 2018) [42]	73.88%	**74.01%**
PIMA	SVM (Sharma & Shah, 2021) [43]	95.6%	**96.8%**
PIMA	LD (Sharma & Shah, 2021) [43]	74%	**80.1%**
PIMA	DT (Iyer & Sumbaly, 2015) [41]	78.24%	**82.03%**
PIMA	NB+DT (Iyer & Sumbaly, 2015) [41]	80.1%	**84.23%**
PIMA	NN (Zou et al., 2018) [42]	**74.14%**	72.6%
PCU	ANN (Nai-arun & Moungmai, 2015) [40]	84.81%	**85.34%**
PIMA	CNN (Yahyaoui et al., 2019) [44]	76.8%	**80.1%**
PIMA	LR (Sharma & Shah, 2021) [43]	76.54%	**77.31%**
PIMA	NB+LR (Sharma & Shah, 2021) [43]	77.01%	**79.31%**
PCU	NB (Nai-arun & Moungmai, 2015) [40]	81.01%	**84.65%**
PCU	LR (Nai-arun & Moungmai, 2015) [40]	82.3%	**86.01%**
PCU	DT (Nai-arun & Moungmai, 2015) [40]	**85.09%**	84.11%
PCU	NB+LR+DT (Nai-arun & Moungmai, 2015) [40]	84.6%	**86.9%**
Compustat	ANFIS (Rahbar, 2022) [45]	92.71%	**94.31%**
Hepatitis	RBF (Novaković, 2015) [46]	78.10%	**80.12%**
LCLDA	KNN (Muslim et al., 2023) [47]	86.69%	**87.21%**

Verification of the obtained values uses statistical criteria. We evaluated the possible normal distribution of the random variable and the difference between the values obtained with and without the proposed method. Verifying the data using the Shapiro-Wilk criterion determines the null hypothesis H0: “The random variable distribution corresponds to the normal law” and the alternative hypothesis H1: “The distribution law is not normal.” For the data groups in Table 1, the values are W = 0.638 and W = 0.690. At n = 18, for a significance level of $p \leq 0.05$, the critical value for the Shapiro-Wilk criterion is W_{cr} = 0.897. $W<W_{cr}$ indicates that hypothesis H0 is rejected. With a high probability, the data distribution is not normal.

Next, two hypotheses are introduced to evaluate statistically significant differences. H0: “$F_1(x)=F_2(x)$, accuracy values obtained without using the proposed method are the same as those obtained with the proposed method; differences can be due to random influences.” H1: “$F_1(x) \neq F_2(x)$, the accuracy values are not the same; differences in the results are significant.” The evaluation of the obtained values in Table 1 used the Wilcoxon criterion. The results of the parameters T_{emp} =12 for $p \leq 0.05$ and n = 18, T_{cr} =40. $T_{emp}<T_{cr}$ shows that the differences are statistically significant. Hypothesis H0 is rejected.

Comparative experiments have shown that the selection of sequences of data sampling segments, in most cases, improves the quality indicators of data processing (Equation 4).

4-4- Analysis of the Results

Increasing the accuracy of classification models is possible by improving the quality of the data coming to the input of algorithms. For this purpose, the clustering of input sequences can be used. Clustering, in some cases, can reduce data scatter, localize outliers, and, to a certain extent, reveal data structures to improve the quality of models. Incorporating information about the belonging of an observation object to a particular cluster provides additional information that can have an impact, especially when there are relatively few predictors in the sample.

Table 1 and the graphs in Figure 6 show that separating the sample into clusters usually improves the quality indicators of the classification algorithms. Clustering produces multiple copies of the classification model, where each copy is trained on data from a single selected cluster. However, there is a threshold in the number of clusters, after which there is no significant improvement in quality indicators. Increasing the number of clusters to a particular limit can obtain quite simple areas of object locations, making it possible to build relatively uncomplicated separating surfaces.

Table 2 and the graphs in Figures 7 and 8 show that the algorithms achieve different values of quality indicators when dividing the sample. Therefore, assigning different classification algorithms to different segments is possible to improve the processing quality. The gain in accuracy ranges from 2% to 9%.

The application of methods aimed at identifying changes in value ranges and event balances allows us to form training samples that locally improve the quality indicators of classification algorithms. However, most experiments show that clustering impacts “weak” processing models more. Table 3 shows that for NB, LR, and DT, the increase in accuracy rate was up to 4%; RF and SVM were up to 1%; and ANN, CNN, ANFIS, RBF, and KNN were up to 2%.

For ensemble methods that use NB+DT, NB+LR, and NB+LR+DT, using the proposed approaches, the increase was up to 4%. However, obtaining the results involved optimization and the setting of classification algorithms for the analyzed data. For NB, data processing was performed to remove correlated features [41]. In DT, depth, maximum number of final nodes, and node separation thresholds were selected [41]. In SVM, kernel function and sensitivity bandwidth settings were determined [43]. In ANN, ANFIS, and CNN, layer selection and setting of membership functions by the backpropagation algorithm were performed [40, 41, 45]. In KNN, the number of neighbors was optimized to construct the separating surface [47]. The presented solution determines the basic parameters without optimization.

Figure 6 shows the increase in accuracy for individual underlying classification algorithms when the number of clusters on model data is increased by up to 7% compared to processing the entire dataset. On the dataset of the compared solutions, the improvement in accuracy was up to 5% because the use of unsupervised clustering makes it possible to localize groups of observation objects and combat various effects, for example, the effect of Simpson's paradox. However, in some cases, when using it, there are problematic issues with data separation. The use of models, methods, and algorithms that separate sequences requires analysis, and changing some parameters of the data sequence fed to the input of the classification algorithm can significantly affect the processing results.

5- Discussion

5-1- Theoretical Contributions

The obtained results further improve the aggregation models of data processing. Using multilevel models makes it possible to reduce the resource intensity of training and retraining algorithms. The proposed paradigm of hierarchy application allows the parallel operation of algorithms, involves actions to coordinate them for achieving the objectives, and analyzes the preconditions used to determine effective models for obtaining results. The hierarchical structure allows the training of the models during operation.

In cases of transformation of input data properties or loss of model adequacy, the underlying algorithms can be promptly substituted. The unsupervised clustering of samples makes it possible to separate the data into relatively small clusters, whose properties are amenable to more qualitative analysis and simpler basic processing models without reducing quality indicators. In contrast to the generally accepted approaches for forming ensemble models, the model under consideration does not require the presence of complex aggregation functions.

5-2- Practical Contribution

Unsupervised clustering makes it possible to estimate the limits of the values of the quality indicators. Training models on clusters provide a preliminary assessment of potential quality indicators and make it possible to determine the processing algorithm for each segment in advance. Reducing the cluster size by increasing their number is reasonable up to a specific limit. After reaching this limit, there is no significant increase in the quality indicators of the data processing models. Assigning algorithms with better quality indicators to segments makes it possible to increase the quality indicators of sample processing compared to separate classifiers and ensemble methods from 2% to 9%.

5-3- Limitations

Despite the relative simplicity of the proposed method, several limitations require consideration when forming processing models. Data samples and information flows have different properties that strongly depend on the subject area. They may have different distributions, statistical indicators of the observation objects, and scatters of values that affect the selection of the most efficient models and algorithms for their processing. Not all models can achieve high-quality indicators of data sampling. During data analysis, data properties may transform: value ranges, data distributions, and the frequency of occurrence of objects of different classes may change. To maintain the adequacy of the models, constant training is necessary. In addition to analyzing the adequacy of the processing models, evaluating the quality of the segmentation and clustering methods is necessary. Various shifts in data values will necessitate the constant evaluation of clustering results. Solving such problems requires constant analysis of not only the data processing models but also the methods of sample separation.

6- Conclusion

This study proposes a multilevel data processing model using sample clustering. The proposed solution involves the ability to build hierarchies, where the upper-level model assigns the most effective lower-level model to a separate cluster, and the middle-level models analyze and generate data. The novelty is the application of a multilevel processing model where, depending on the data properties, the selection of the base model with the best quality indicators occurs. Data separation aims to improve the quality indicators of model processing by identifying potential data structures and reducing the scatter of parameters inside individual segments. The formation of multilevel structures that process, analyze incoming information flows, and assign the most appropriate model for solving the current problem can reduce the complexity and resource intensity of classical ensemble methods, deal with overtraining, reduce dependence on base models, increase the efficiency of adjustment of underlying algorithms in transforming data properties, and improve the interpretability of results. The proposed solution aims to further improve and extend ensemble methods.

7- Declarations

7-1- Author Contributions

Conceptualization, I.L. and M.S.; methodology, I.L.; software, I.L.; validation, I.L. and M.S.; investigation, I.L. and M.S.; resources, M.S.; data curation, I.L.; writing—original draft preparation, M.S.; writing—review and editing, I.L.; visualization, M.S.; supervision, I.L.; project administration, I.L.; funding acquisition, M.S. All authors have read and agreed to the published version of the manuscript.

7-2- Data Availability Statement

The data presented in this study are available in the present article.

7-3- Funding

The authors received no financial support for the research, authorship, and/or publication of this article.

7-4- Institutional Review Board Statement

Not applicable.

7-5- Informed Consent Statement

Not applicable.

7-6- Conflicts of Interest

The authors declare that there is no conflict of interest regarding the publication of this manuscript. In addition, the ethical issues, including plagiarism, informed consent, misconduct, data fabrication and/or falsification, double publication and/or submission, and redundancies have been completely observed by the authors.

8- References

[1] Wu, Y., Zhao, R., Zhu, J., Chen, F., Xu, M., Li, G., Song, S., Deng, L., Wang, G., Zheng, H., Ma, S., Pei, J., Zhang, Y., Zhao, M., & Shi, L. (2022). Brain-inspired global-local learning incorporated with neuromorphic computing. Nature Communications, 13(1), 65. doi:10.1038/s41467-021-27653-2.

[2] Mohammed, A., & Kora, R. (2023). A comprehensive review on ensemble deep learning: Opportunities and challenges. Journal of King Saud University - Computer and Information Sciences, 35(2), 757–774. doi:10.1016/j.jksuci.2023.01.014.

[3] de Zarzà, I., de Curtò, J., Hernández-Orallo, E., & Calafate, C. T. (2023). Cascading and Ensemble Techniques in Deep Learning. Electronics (Switzerland), 12(15), 3354. doi:10.3390/electronics12153354.

[4] Mienye, I. D., & Sun, Y. (2022). A Survey of Ensemble Learning: Concepts, Algorithms, Applications, and Prospects. IEEE Access, 10, 99129–99149. doi:10.1109/ACCESS.2022.3207287.

[5] Akano, T. T., & James, C. C. (2022). An assessment of ensemble learning approaches and single-based machine learning algorithms for the characterization of undersaturated oil viscosity. Beni-Suef University Journal of Basic and Applied Sciences, 11(1), 149. doi:10.1186/s43088-022-00327-8.

[6] Mishra, S., Shaw, K., Mishra, D., Patil, S., Kotecha, K., Kumar, S., & Bajaj, S. (2022). Improving the Accuracy of Ensemble Machine Learning Classification Models Using a Novel Bit-Fusion Algorithm for Healthcare AI Systems. Frontiers in Public Health, 10. doi:10.3389/fpubh.2022.858282.

[7] Valencia-Vidal, B., Ros, E., Abadía, I., & Luque, N. R. (2023). Bidirectional recurrent learning of inverse dynamic models for robots with elastic joints: a real-time real-world implementation. Frontiers in Neurorobotics, 17. doi:10.3389/fnbot.2023.1166911.

[8] Zhang, Y., Liu, J., & Shen, W. (2022). A Review of Ensemble Learning Algorithms Used in Remote Sensing Applications. Applied Sciences (Switzerland), 12(17), 8654. doi:10.3390/app12178654.

[9] Trevizan, B., Chamby-Diaz, J., Bazzan, A. L. C., & Recamonde-Mendoza, M. (2020). A comparative evaluation of aggregation methods for machine learning over vertically partitioned data. Expert Systems with Applications, 152, 113406. doi:10.1016/j.eswa.2020.113406.

[10] Wang, S., Zhou, W., & Jiang, C. (2020). A survey of word embeddings based on deep learning. Computing, 102(3), 717–740. doi:10.1007/s00607-019-00768-7.

[11] Vousden, M., Morris, J., McLachlan Bragg, G., Beaumont, J., Rafiev, A., Luk, W., Thomas, D., & Brown, A. (2023). Event-based high throughput computing: A series of case studies on a massively parallel softcore machine. IET Computers and Digital Techniques, 17(1), 29–42. doi:10.1049/cdt2.12051.

[12] Huang, J., Chen, P., Lu, L., Deng, Y., & Zou, Q. (2023). WCDForest: a weighted cascade deep forest model toward the classification tasks. Applied Intelligence, 53(23), 29169–29182. doi:10.1007/s10489-023-04794-z.

[13] Brown, A. D., Beaumont, J. R., Thomas, D. B., Shillcock, J. C., Naylor, M. F., Bragg, G. M., Vousden, M. L., Moore, S. W., & Fleming, S. T. (2023). POETS: An Event-driven Approach to Dissipative Particle Dynamics. ACM Transactions on Parallel Computing, 10(2), 1–32. doi:10.1145/3580372.

[14] Marques, H. O., Swersky, L., Sander, J., Campello, R. J. G. B., & Zimek, A. (2023). On the evaluation of outlier detection and one-class classification: a comparative study of algorithms, model selection, and ensembles. Data Mining and Knowledge Discovery, 37(4), 1473–1517. doi:10.1007/s10618-023-00931-x.

[15] Huang, W., & Ding, N. (2021). Privacy-Preserving Support Vector Machines with Flexible Deployment and Error Correction. In: Deng, R., et al. Information Security Practice and Experience. ISPEC 2021. Lecture Notes in Computer Science, 13107. Springer, Cham, Switzerland. doi:10.1007/978-3-030-93206-0_15.

[16] Liu, N., & Zhao, J. (2023). Streaming Data Classification Based on Hierarchical Concept Drift and Online Ensemble. IEEE Access, 11, 126040–126051. doi:10.1109/ACCESS.2023.3327637.

[17] Xu, H., Zhang, Y., Zhou, B., Wang, L., Yao, X., Meng, G., & Shen, S. (2022). Omni-Swarm: A Decentralized Omnidirectional Visual-Inertial-UWB State Estimation System for Aerial Swarms. IEEE Transactions on Robotics, 38(6), 3374–3394. doi:10.1109/TRO.2022.3182503.

[18] Zhang, X., & Wang, M. (2021). Weighted Random Forest Algorithm Based on Bayesian Algorithm. Journal of Physics: Conference Series, 1924(1), 12006. doi:10.1088/1742-6596/1924/1/012006.

[19] Colter, Z., Fayazi, M., Youbi, Z. B. El, Kamp, S., Yu, S., & Dreslinski, R. (2022). Tablext: A combined neural network and heuristic based table extractor. Array, 15, 100220. doi:10.1016/j.array.2022.100220.

[20] Di Franco, G., & Santurro, M. (2021). Machine learning, artificial neural networks and social research. Quality & Quantity, 55(3), 1007–1025. doi:10.1007/s11135-020-01037-y.

[21] Piernik, M., & Morzy, T. (2021). A study on using data clustering for feature extraction to improve the quality of classification. Knowledge and Information Systems, 63(7), 1771–1805. doi:10.1007/s10115-021-01572-6.

[22] ChauPattnaik, S., Ray, M., & Nayak, M. M. (2021). Component based reliability prediction. International Journal of System Assurance Engineering and Management, 12(3), 391–406. doi:10.1007/s13198-021-01079-x.

[23] Si, S., Zhao, J., Cai, Z., & Dui, H. (2020). Recent advances in system reliability optimization driven by importance measures. Frontiers of Engineering Management, 7(3), 335–358. doi:10.1007/s42524-020-0112-6.

[24] Djouzi, K., Beghdad-Bey, K., & Amamra, A. (2022). A new adaptive sampling algorithm for big data classification. Journal of Computational Science, 61, 101653. doi:10.1016/j.jocs.2022.101653.

[25] Lebedev, I. S., & Sukhoparov, M. E. (2023). Adaptive Learning and Integrated Use of Information Flow Forecasting Methods. Emerging Science Journal, 7(3), 704–723. doi:10.28991/ESJ-2023-07-03-03.

[26] Sugita, I., Matsuyama, S., Dobashi, H., Komura, D., & Ishikawa, S. (2022). Viola: a structural variant signature extractor with user-defined classifications. Bioinformatics, 38(2), 540–542. doi:10.1093/bioinformatics/btab662.

[27] Peruvemba Ramaswamy, V., & Szeider, S. (2021). Turbocharging Treewidth-Bounded Bayesian Network Structure Learning. Proceedings of the AAAI Conference on Artificial Intelligence, 35(5), 3895–3903. doi:10.1609/aaai.v35i5.16508.

[28] Debnath, S., Arif, W., Roy, S., Baishya, S., & Sen, D. (2022). A Comprehensive Survey of Emergency Communication Network and Management. Wireless Personal Communications, 124(2), 1375–1421. doi:10.1007/s11277-021-09411-1.

[29] Adeen, N., Abdulazeez, M., & Zeebaree, D. (2020). Systematic review of unsupervised genomic clustering algorithms techniques for high dimensional datasets. Technol. Reports Kansai University, 62(3), 355-374.

[30] Saraçoğlu, R., & Nemati, N. (2020). Vehicle Detection Using Fuzzy C-Means Clustering Algorithm. International Journal of Applied Mathematics Electronics and Computers, 8(3), 85–91. doi:10.18100/ijamec.799431.

[31] Sri, K. S., Nayaka, R. R., & Kumar, M. V. N. S. (2023). Mechanical properties of sustainable self-healing concrete and its performance evaluation using ANN and ANFIS models. Journal of Building Pathology and Rehabilitation, 8(2), 99. doi:10.1007/s41024-023-00345-8.

[32] Xu, S., Song, Y., & Hao, X. (2022). A Comparative Study of Shallow Machine Learning Models and Deep Learning Models for Landslide Susceptibility Assessment Based on Imbalanced Data. Forests, 13(11), 1908. doi:10.3390/f13111908.

[33] Mehrabi, M., Pradhan, B., Moayedi, H., & Alamri, A. (2020). Optimizing an adaptive neuro-fuzzy inference system for spatial prediction of landslide susceptibility using four state-of-the-art metaheuristic techniques. Sensors (Switzerland), 20(6), 1723. doi:10.3390/s20061723.

[34] Wei, A., Yu, K., Dai, F., Gu, F., Zhang, W., & Liu, Y. (2022). Application of Tree-Based Ensemble Models to Landslide Susceptibility Mapping: A Comparative Study. Sustainability (Switzerland), 14(10), 6330. doi:10.3390/su14106330.

[35] Ji, X., Liu, S., Zhao, P., Li, X., & Liu, Q. (2021). Clustering Ensemble Based on Sample's Certainty. Cognitive Computation, 13(4), 1034–1046. doi:10.1007/s12559-021-09876-z.

[36] Zhong, G., Shu, T., Huang, G., & Yan, X. (2022). Multi-view spectral clustering by simultaneous consensus graph learning and discretization. Knowledge-Based Systems, 235, 107632. doi:10.1016/j.knosys.2021.107632.

[37] Tong, W., Wang, Y., & Liu, D. (2023). An Adaptive Clustering Algorithm Based on Local-Density Peaks for Imbalanced Data Without Parameters. IEEE Transactions on Knowledge and Data Engineering, 35(4), 3419–3432. doi:10.1109/TKDE.2021.3138962.

[38] He, H., Liu, W., Zhao, Z., He, S., & Zhang, J. (2022). Vulnerability of Regional Aviation Networks Based on DBSCAN and Complex Networks. Computer Systems Science and Engineering, 43(2), 643–655. doi:10.32604/csse.2022.027211.

[39] Tkachenko, R. (2022). An Integral Software Solution of the SGTM Neural-Like Structures Implementation for Solving Different Data Mining Tasks. Lecture Notes in Computational Intelligence and Decision Making, ISDMCI 2021, Lecture Notes on Data Engineering and Communications Technologies, 77, Springer, Cham, Switzerland. doi:10.1007/978-3-030-82014-5_48.

[40] Nai-Arun, N., & Moungmai, R. (2015). Comparison of Classifiers for the Risk of Diabetes Prediction. Procedia Computer Science, 69, 132–142. doi:10.1016/j.procs.2015.10.014.

[41] Iyer, A., S, J., & Sumbaly, R. (2015). Diagnosis of Diabetes Using Classification Mining Techniques. International Journal of Data Mining & Knowledge Management Process, 5(1), 01–14. doi:10.5121/ijdkp.2015.5101.

[42] Zou, Q., Qu, K., Luo, Y., Yin, D., Ju, Y., & Tang, H. (2018). Predicting Diabetes Mellitus with Machine Learning Techniques. Frontiers in Genetics, 9. doi:10.3389/fgene.2018.00515.

[43] Sharma, T., & Shah, M. (2021). A comprehensive review of machine learning techniques on diabetes detection. Visual Computing for Industry, Biomedicine, and Art, 4, 30. doi:10.1186/s42492-021-00097-7.

[44] Yahyaoui, A., Jamil, A., Rasheed, J., & Yesiltepe, M. (2019). A Decision Support System for Diabetes Prediction Using Machine Learning and Deep Learning Techniques. 2019 1st International Informatics and Software Engineering Conference (UBMYK), Ankara, Turkey. doi:10.1109/ubmyk48245.2019.8965556.

[45] Rahbar, M. A. (2022). Evaluation of the hybrid method of genetic algorithm and adaptive neural-fuzzy network (ANFIS) model in predicting the bankruptcy of companies listed on the Tehran stock exchange. Journal of Applied Research on Industrial Engineering, 9(3), 274–290. doi:10.22105/jarie.2021.254142.1204.

[46] Novakovic, J. D. (2015). Estimating Performances of Learned Knowledge for the RBF Network as an Artificial Intelligence Method. Strategic Management, 20(4), 46–53.

[47] Muslim, M. A., Nikmah, T. L., Pertiwi, D. A. A., Subhan, Jumanto, Dasril, Y., & Iswanto. (2023). New model combination meta-learner to improve accuracy prediction P2P lending with stacking ensemble learning. Intelligent Systems with Applications, 18, 200204. doi:10.1016/j.iswa.2023.200204.