Complex Systems Informatics and Modeling Quarterly (CSIMQ)
eISSN: 2255-9922
Published online by RTU Press, https://csimq-journals.rtu.lv
Article 221, Issue 40, September/October 2024, Pages 94–110
https://doi.org/10.7250/csimq.2024-40.04

Machine Learning Analysis of Arterial Oscillograms for Depression Level Diagnosis in Cardiovascular Health

Vladislav Kaverinsky[1], Dmytro Vakulenko[2], Liudmyla Vakulenko[3], and Kyrylo Malakhov[4]*

[1] Frantsevic Institute for Problems in Material Science of the National Academy of Sciences of Ukraine, Kyiv, Ukraine
[2] Medical Informatics Department, Horbachevsky Ternopil National Medical University, Ternopil, Ukraine
[3] Ternopil Volodymyr Hnatiuk National Pedagogical University, Ternopil, Ukraine
[4] Microprocessor technology lab, Glushkov Institute of Cybernetics of the National Academy of Sciences of Ukraine, Kyiv, Ukraine

insamhlaithe@gmail.com, vakulenko@tdmu.edu.ua, vakulenko3@ukr.net, k.malakhov@incyb.kiev.ua

Abstract. The presented study explores the clustering of arterial oscillogram (AO) data among a sample of patients, focusing on ultra-low-frequency (ULF) indicators and their relationship with depression levels. Through dimensionality reduction using UMAP, two distinct classes emerged, categorized as lighter and more severe cases. Utilizing machine learning methods, an automated classifier was developed based on correlated ULF indicators, which led to improved classification accuracy. By incorporating ULF parameters, products of correlated parameters, and additional measured factors, the classifier achieved high reliability in estimating depression levels. Specifically, the nearest neighbors method yielded accuracies up to 0.9792. This research supports the creation of an automated diagnostic classification AI service capable of reliably estimating at least four levels of depression based on AO analysis.

Keywords: Machine Learning, Transdisciplinary Research, Data Clustering, UMAP, Arterial Oscillogram, ULF, Mental State Diagnostic.

1 Introduction

Measurement of blood pressure is a mandatory procedure in the activity of a doctor at all stages of medical care [1], [2]. The use of electronic blood pressure meters made it possible to improve and expand the informativeness of the blood pressure measurement process. Thus, some special models of electronic pulsimeters make it possible to use registered arterial pulsations not only to

* Corresponding author

Reference: V. Kaverinsky, D. Vakulenko, L. Vakulenko, and K. Malakhov, “Machine Learning Analysis of Arterial Oscillograms for Depression Level Diagnosis in Cardiovascular Health,” Complex Systems Informatics and Modeling Quarterly, CSIMQ, no. 40, pp. 94–110, 2024. Available: https://doi.org/10.7250/csimq.2024-40.04

Additional information. Author ORCID iD: V. Kaverinsky – https://orcid.org/0000-0002-6940-579X, D. Vakulenko – https://orcid.org/0000-0001-5062-9278, L. Vakulenko – https://orcid.org/0000-0002-4723-3747, and K. Malakhov – https://orcid.org/0000-0003-3223-9844. PII S2255992224000221X. Article received: 3 May 2024. Last revision received: 17 October 2024. Accepted: 24 October 2024. Available online: 31 October 2024.

calculate blood pressure values but also to transmit them for further analysis [3]. The obtained values contain information that can give a deeper insight into the general state of the body and its systems [1], [4]–[6]. Then, in the calculation service, arterial oscillograms are formed from the blood pressure curve, which is subject to further analysis.

The practical implementation of the arterial oscillography method opens up a wide field of activity for both scientists and practical medicine. Including, it provides opportunities for using machine learning methods to build automated diagnostic AI classification systems capable of determining the patient's condition based on the objective results of the specified measurements. In particular, this work is aimed at the possibility of building such a classifier that would be able to estimate the level of depression, not simply based on the results of the patient survey, which may be biased and taken before, but specifically based on the results of measuring blood pressure oscillograms, which, as noted, can act as markers of the state of the body. The relation between the oscillograms data and the level of depression for the moment is mostly hypothetical. The aim of the presented research is an attempt to find out some corresponding interrelations, which could be helpful in the diagnostic classification system development. The primary attention is to be played (but not limited to) the ultra-low-frequency (ULF) oscillations, which have been found to coordinate the functional activity of all body systems following changes in the external environment [2].

The article is organized as follows. Section 2 describes the background of the research and related works. Section 3 and Section 4 give details regarding indicators and data, respectively, used for machine learning. The research methodology is discussed in Section 5. The dimension reduction is concerned in Section 6. The created classifier is presented in Section 7. Brief conclusions are provided in Section 8.

2 Background and Related Works

The development of modern technologies has had a profound impact on the realm of intellectual activities, particularly in the field of scientific research and development. In this context, a new class of information systems has emerged – the Research and Development Workstation Environment (RDWE) [7], which implements an enhanced concept of an automated workstation for ongoing research and related intellectual information technologies. These systems and concepts encompass the main stages of the life cycle of scientific research and development – from semantic analysis of information materials from various subject areas to the development of constructive features of innovative proposals. A notable feature of RDWE systems is their adaptability (problem orientation) to various types of scientific activities, achieved through the integration of diverse functional services and the ability to add new ones within a hybrid cloud environment/platform.

Among the most impressive examples of modern RDWE systems is the automated interactive system OntoChatGPT [8]. This system is developed using advanced computational linguistics technologies, such as GPT-4 from OpenAI, support services for ontological engineering, and natural language understanding. A detailed overview of the RDWE system OntoChatGPT and its evolution can be found in studies [9]–[12].

In early 2022, the research project named "Development of the cloud-based platform for patient-centered telerehabilitation of oncology patients with mathematical-related modeling" [13], [14] was established in Ukraine. The project is dedicated to the development of a hybrid cloud platform and the creation of information technology for telerehabilitation of cancer patients, serving a wide range of specialists in physical and rehabilitation medicine in the "Telerehabilitation of Cancer Patients" subject domain. The research presented in this study was conducted utilizing the developed hybrid cloud environment/platform for telerehabilitation medicine [15]–[17] as part of the mentioned project.

Machine learning algorithms are crucial tools for analyzing complex datasets, finding patterns, and making predictions. These algorithms vary in their approaches and performance based on the nature of the data and the task at hand. In clustering and classification tasks, several models, such as UMAP, KNeighborsClassifier, XGBClassifier, and LGBMClassifier, are widely used. When addressing clustering and classification tasks, these algorithms can often be combined to achieve optimal results. A typical workflow might involve using Uniform Manifold Approximation and Projection (UMAP) to reduce the dimensionality of a high-dimensional dataset. This reduction helps to identify clusters and patterns that are not easily detectable in the original feature space. UMAP's ability to compress data while retaining both global and local structures makes it feasible to prepare data for classification models. After UMAP has revealed the structure of the data, classifiers can be applied. For instance, k-NN is useful for straightforward classification tasks where proximity in feature space is indicative of class labels. For more complex datasets, where non-linear relationships exist, gradient-boosting models like XGBClassifier or LGBMClassifier may provide superior performance due to their ability to capture intricate patterns through ensemble learning.

The UMAP is a powerful algorithm for dimensionality reduction, clustering, and data visualization. UMAP is particularly effective in handling high-dimensional data where it compresses the features into lower dimensions while preserving the global structure and local relationships of the data point [18]. UMAP works by constructing a weighted graph representing the high-dimensional manifold. It optimizes a low-dimensional layout by maintaining the closest neighbors' pairwise distances while simultaneously distributing non-neighbors more uniformly. This results in a low-dimensional projection of data that reveals clusters and substructures. One of the key strengths of UMAP is its speed and scalability, which make it suitable for large datasets. In clustering tasks, UMAP is often employed as a preprocessing step. By reducing the dimensionality, it helps perform classification tasks more efficiently.

The KNeighborsClassifier is one of the simplest machine learning algorithms, yet it is highly effective for classification tasks. It operates based on the principle of proximity. When presented with a new data point, the algorithm identifies the 'k' closest training examples in the feature space and assigns the most common class label among those neighbors to the new instance. KNeighborsClassifier is a non-parametric model, meaning it makes no assumptions about the underlying distribution of the data. Its performance heavily depends on the choice of 'k' (the number of neighbors) and the distance metric (e.g., Euclidean distance). Despite its simplicity, this algorithm can be computationally expensive, especially when dealing with large datasets, as it needs to compute the distance between the new data point and every point in the training set [19], [20].

The XGBClassifier is an implementation of the extreme gradient boosting (XGBoost) algorithm, which is a type of decision-tree-based ensemble method. XGBoost improves the performance of classical decision trees by optimizing through gradient boosting, where models are added iteratively to correct the errors of previous ones. The core idea behind gradient boosting is to construct a strong learner by combining several weak learners. In XGBClassifier, each tree in the ensemble minimizes a loss function, typically via gradient descent, adjusting its parameters to make better predictions. This iterative process continues until the algorithm achieves a predefined accuracy or another stopping criterion is met. One of the key advantages of XGBClassifier is its regularization ability, which helps prevent overfitting while maintaining high accuracy [21].

The LGBMClassifier is another gradient boosting algorithm but with several optimizations that make it faster and lighter than XGBoost. LGBM, or LightGBM, splits trees leaf-wise rather than level-wise (as in XGBoost), which leads to more accurate splits and faster convergence. LGBMClassifier also implements techniques like histogram-based decision-making which significantly reduces the training time. One of the most notable features of LGBMClassifier is its ability to handle large-scale datasets with millions of rows and numerous features. Its leaf-wise splitting method enables it to capture more complex patterns in data, making it highly effective in real-world tasks like classification, regression, and ranking [22].

Recent advancements in machine learning have significantly contributed to the development of diagnostic tools for mental health assessment, particularly in the realm of cardiovascular indicators like arterial oscillograms (AOs). Various studies have utilized machine learning algorithms to classify and analyze cardiovascular data for mental health insights, achieving notable accuracy improvements in the process.

One key approach in the field has been leveraging clustering and dimensionality reduction techniques to process large datasets, as demonstrated in [23]. This study utilized clustering methods to segment cardiovascular data and improve the diagnosis of mental health conditions by focusing on specific low-frequency oscillations in heart rate variability (HRV). Such approaches align with research indicating that specific frequency components of HRV correlate with mental health status, making them effective in predictive models.

Another significant contribution by Geng et al. [24] explored the application of advanced classification techniques, such as neural networks, in identifying depression from cardiovascular data. Their research highlights the effectiveness of deep learning models, which can accurately classify multiple depression levels by interpreting complex physiological patterns. This study's findings underscore the potential for these models to surpass traditional methods in diagnostic precision.

Additionally, Xia et al. [25] employed feature engineering techniques to refine the input parameters for cardiovascular data classifiers, increasing diagnostic accuracy. Their methodology incorporated cross-validation of feature sets, ultimately improving the robustness of predictions of a depression level. By isolating the most influential features from hundreds of data points, their work offers insights into optimizing machine learning models for mental health diagnostics.

Similarly, recent work by Yang et al. [26] utilized unsupervised machine learning to identify latent patterns within AO datasets, demonstrating that multi-modal clustering can significantly enhance the identification of depression severity. Their approach involved UMAP for dimensionality reduction, which was instrumental in creating more defined clusters corresponding to different mental health states.

The integration of machine learning techniques with cardiovascular diagnostics offers a promising avenue for developing accessible and reliable tools for mental health assessment. Ongoing research continues to refine these methods, focusing on improving accuracy, interpretability, and applicability in clinical settings.

3 Characteristics of Measured Indicators for Input Data Formation

The resulting oscillograms were subjected to temporal and frequency (spectral) analyses [27]. The duration of arterial pulsations at positive and negative extremes was studied. The spectral analysis determined the spectral power of the interferogram (45 indicators), the arterial oscillogram itself using Fourier transformation (340 indicators), and the instantaneous frequency and phase using Hilbert-Huang transformation (90 indicators) [3]. The following indicators were used to analyze interferograms [28], [29]:

- **TR** – a full spectrum of frequencies, indicating the influence of both sympathetic and parasympathetic parts of the nervous system.
- **HF** – the power of the high-frequency component (0.15–0.40 Hz), which reflects parasympathetic influence.
- **LF –** the power of the low-frequency component (0.04–0.15 Hz), indicating the activity of the vasomotor center and reflecting sympathetic and parasympathetic influences from above the peripheral level to the centers of autonomic innervation in the medulla oblongata. LF/HF is the ratio of low- and high-frequency waves, representing vegetative balance.
- **VLF** – the power of the very low-frequency component (0.003–0.04 Hz). The functional value of VLF waves is related to hormonal and metabolic effects on heart rhythm, reflecting the

influence of higher autonomic centers on the cardiovascular subcortical center and the higher brain regions.

- **ULF** – the power of ultra-low-frequency (ultra-slow) oscillations (with a frequency of less than 0.003 Hz). The functional value of ULF waves involves integrating and adapting the organism's functional state to external factors, coordinating the functional activity of all body systems in response to environmental changes.

4 Initial Dataset and Numerical Research Procedure

The patients with mental disorders, aged 32–65, from whom data were obtained, were treated at the Ternopil Regional Clinical Psychoneurological Hospital. The degree of mental and psychotic disorders was assessed using the Hospital Depression Rating Scale (HDRS) and the DASS-21 depression, anxiety, and stress scale. The main spectrum of diagnoses included bipolar affective disorder with a current episode of depression and depressive disorders without psychotic inclusions, presenting with a depressive syndrome of varying severity.

The groups of input parameters were analyzed separately because having 1030 factors with a comparably short dataset (181 data rows only) could lead to distorted results. Each data row corresponds to a specific patient. All patients were divided into 5 a priori groups according to the primary estimated depression level. The first 4 groups were formed based on questionnaire results, with the estimated depression level increasing from the 1st to the 4th. Patients receiving treatment in the mental hospital were included in the 5th group.

Data normalization is fundamental in many machine learning tasks because it ensures that all features contribute equally to the model's performance. In datasets, features often have different scales, and the provided dataset is no exception. Algorithms like k-nearest neighbors or gradient boosting models rely on distance metrics to determine relationships between data points. If features with larger ranges dominate, they can skew results, leading to suboptimal model performance. Normalization scales features to a common range, ensuring that each one has an equal impact. Moreover, keeping feature values on a similar scale helps these models optimize their loss functions more efficiently, preventing erratic jumps during gradient descent and leading to faster and more reliable convergence [30]. Therefore, all analyses were carried out on the normalized data. The normalized value for each feature was calculated as follows:

$$a_{i\_norm} = \frac{a_i - a_{\min}}{a_{\max} - a_{\min}} \tag{1}$$

where a_i is each current value; $a_{\min}$ is the minimum value of the column; $a_{\max}$ is the maximum value of the column.

So, at $a_i = a_{\min}$, the normalized value becomes 0 and, at $a_i = a_{\max}$, it turns to 1. Thus, all the working values will be in the range of [0; 1].

Mapping to the original values was also saved.

For the reuse possibility of the developed classifiers, the normalization range could be expanded to avoid the cases when new data values exceed their initial range.

The dataset included in some cases zero and omitted (None) values. For such factors as "systole", "diastole", "pressure", "M1", "M2", "M3", "M4", "M5", "M6", "M7", "M8", "M9", and all the Hurst parameters they were replaced with a mean value estimated excluding zeros and "Nones". All the other factors were treated as equal to 0 and were neither replaced nor removed.

Visual cluster analysis and dimensionality reduction are crucial techniques for understanding complex, high-dimensional datasets. Dimensionality reduction methods like UMAP project high-dimensional data into lower-dimensional spaces, making patterns and clusters more interpretable through visualization. This process helps reveal hidden structures, relationships, or anomalies that may be otherwise obscured in higher dimensions. Visualization facilitates the detection of clusters, enabling us to identify natural groupings in the data. This is especially useful in tasks like exploratory data analysis (EDA), where visual insights can inform feature engineering and algorithm selection. Additionally, by reducing the number of dimensions, dimensionality reduction techniques help reduce computational costs and avoid situations where model performance deteriorates with too many features [18], [31]. To perform a visual cauterization analysis, dimensionality reduction was used using UMAP method implemented in the appropriate Python library [32].

For the automatic separation of the classes, “KNeighborsClassifier” from the Scikit Learn library was used. It implements the method of KNeighbours. The classifier was applied to the UMAP embeddings space previously obtained by the described above method. To investigate the classifiers working with dataset selections without UMAP the dimensionality reduction “Lazy Predict” technique from the Scikit Learn framework was used. It provides automatic learning of a several types of classifiers included in Scikit Learn with the default settings on the giving learning sample and then tests them on the test sample. As a result, it returns the list of the studied classifiers ranged according their obtained accuracy values.

5 Research Methodology

The dataset under consideration exhibits a notable characteristic, namely, an extensive array of parameters exceeding a thousand in number, juxtaposed with a significantly smaller volume of data rows corresponding to 181 patients. Moreover, a substantial portion of these parameters demonstrates inter-correlations. It is unfeasible to utilize such a dataset in its raw state for constructing a reliable classifier. Consequently, a parameter selection process was imperative to identify the most influential features essential for effective class differentiation. Additionally, a key challenge was ascertaining whether the predefined classes exhibited clear separability or if their boundaries were indistinct.

The clusters of features with clearly defined boundaries (i.e., those where the classes show a discernible separation) are prioritized for building the classifier. This approach is essential for improving the classifier’s accuracy by ensuring it operates on features that are strongly correlated with the target classes. Conversely, classes with less distinct boundaries were initially merged into macro classes to simplify the classification task at this stage. Incorporating feature correlations further refines the classifier. Correlations within specific classes or across subsets of classes, but not others, act as distinguishing factors. This observation is supported in [33], where is emphasized that interdependencies between features can be exploited to improve classification performance. The inclusion of correlated feature products expands the feature set helping to capture complex relationships within the data.

A general scheme of the research procedure (six stages) is shown in Figure 1. Given that the dataset parameters were initially categorized into distinct groups (stage 1), each with its explanatory context, an initial analysis treated these groups separately. This preliminary investigation primarily focused on *data clustering* within a reduced-dimensional space to align the resulting clusters' structure with the data distribution among them (stage 2). An integral aspect at this stage was identifying the feature group capable of yielding clusters that closely approximated the a priori class separations. Consequently, this feature group formed the foundation for the classifier, with priority given to classes exhibiting discernible separation within the clusters manifold. Conversely, classes with blurry boundaries were amalgamated into macro classes, at least during the initial phases.

However, not all features within the selected group were inherently valuable for classification. Furthermore, practical experience indicated that the direct utilization of raw data values did not yield a classifier of satisfactory quality. Therefore, an important characteristic of the employed method involved *accounting for feature correlations* within each proposed class (stage 3). It was observed that certain features exhibited strong correlations within specific classes (or across several, but not all classes), whereas these correlations were notably weaker or absent in other classes. Additionally, theoretical considerations suggested that *features in different classes might display varying correlation signs*, serving as potential distinguishing factors

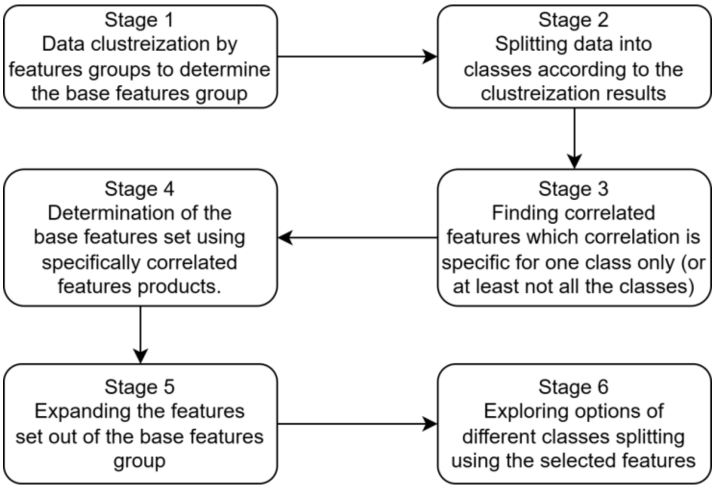

Stage 1
Data clustreization by features groups to determine the base features group
Stage 2
Splitting data into classes according to the clustreization results
Stage 3
Finding correlated features which correlation is specific for one class only (or at least not all the classes)
Stage 4
Determination of the base features set using specifically correlated features products.
Stage 5
Expanding the features set out of the base features group
Stage 6
Exploring options of different classes splitting using the selected features

Figure 1. General scheme of the research process

To enhance classification accuracy, the input parameters for the classifier were selected based on a combination of features and their value products, particularly focusing on features with correlated patterns across classes (stage 4). Subsequently, attempts were made to *expand the input feature set* (stage 5) beyond the base feature group by identifying distinct correlations between features from the base group and others. This process involved selecting a few highly effective additional features that significantly improved classification accuracy, guided by a systematic search methodology, shown in Figure 2.

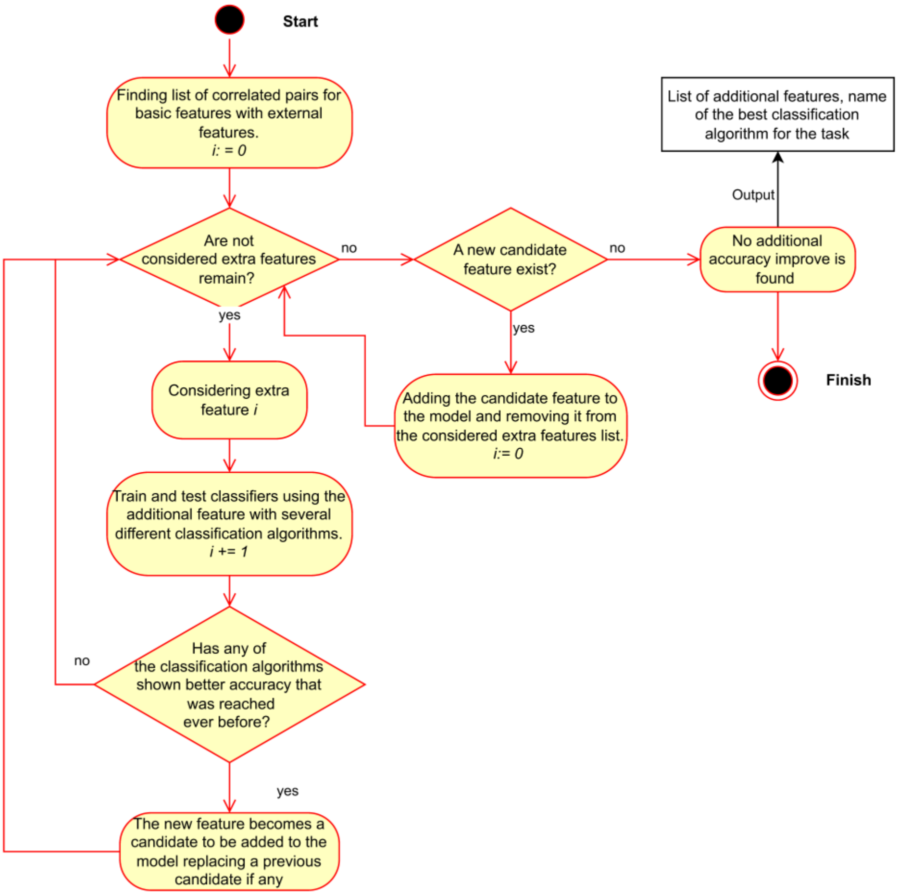

Start
Finding list of correlated pairs for basic features with external features.
i: = 0
Are not considered extra features remain?
no
yes
A new candidate feature exist?
no
yes
No additional accuracy improve is found
Output
List of additional features, name of the best classification algorithm for the task
Finish
Considering extra feature i
Adding the candidate feature to the model and removing it from the considered extra features list.
i:= 0
Train and test classifiers using the additional feature with several different classification algorithms.
i += 1
Has any of the classification algorithms shown better accuracy that was reached ever before?
no
yes
The new feature becomes a candidate to be added to the model replacing a previous candidate if any

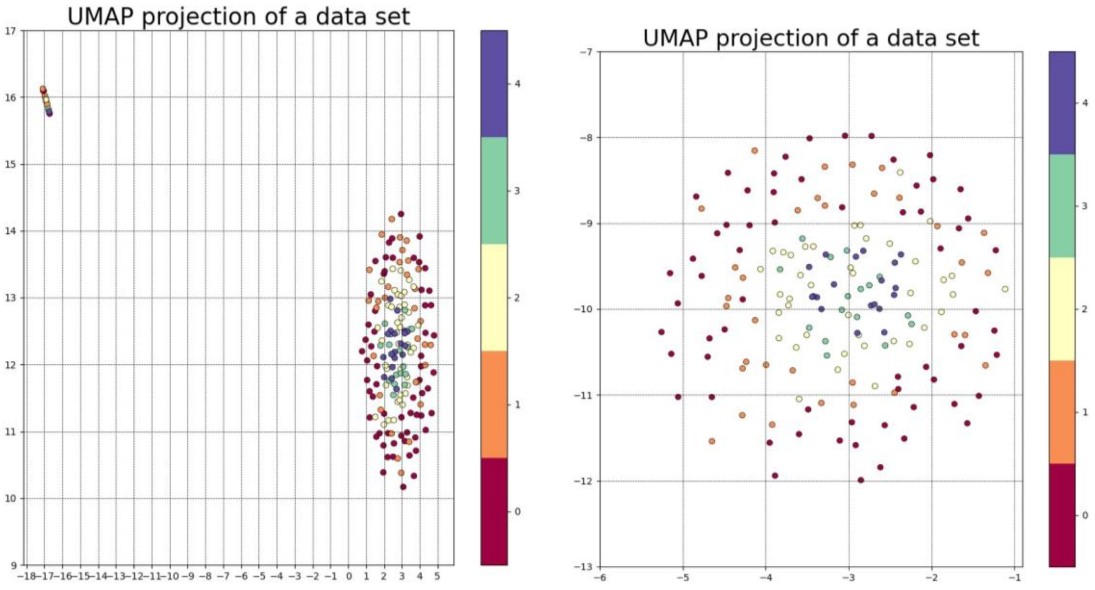
Figure 2. Finding extra features for the classification model

For this purpose, distinguishable correlations were found between the features from the base group and all the others. Because the number of features is quite large, there may be rather a lot of such additional features. Not all of them are equally effective for the classification task, and only a few of the most valuable ones are to be selected. For the selection, the following search methodology was used: to the previously determined feature set, a new one, which is a product of values from one of the found correlated pairs, is added; then the classifier with this new feature is retrained; if it is possible to obtain, through any of the considered classification algorithms, a better accuracy value than was previously possible, the pair product becomes a candidate to be added to the classification model. However, other pairs are also considered, and if some of them increase the accuracy to a value that has never been reached before, it becomes a new candidate to be added to the classification model, replacing the previous candidate. When all the options are considered, the paired product that mostly increases the accuracy is added to the model. Then the algorithm is restarted, attempting in this way to find another pair product that is able to increase the accuracy. The searching process is stopped when one of the following conditions is reached: the addition of any new feature does not increase the classification accuracy, the accuracy reaches the desired values, the number of features reaches a set limit, or there are no new features to add (a hypothetical case if the number of such extra features is small).

At the final stage (stage 6), *using the selected feature set, various classification and separation options were explored*, aiming to achieve satisfactory results. This approach facilitated the development of a complex classifier capable of delineating different class groups, thereby enhancing the ability to detect multiple classes within the dataset

6 UMAP Embeddings Clusterization to Study the Possibilities of Cogent and Reasonable Classification

UMAP transformation with different metrics has been used for several presented factor groups. However, a picture, consistent with the a priori given classes, was observed for the cases of Jacquard metric being applied to the factors group corresponding to the Fourier transform spectrum powers of the ultralow (<0.003 Hz) frequencies (ULF) that revealed an interesting result. It is shown in Figure 3 (a). Other metrics either did not provide certain clusters, or the clusters obtained were not in accordance with the a priori patient groups.



a)

b)

Figure 3. UMAP transformation representation of the ULF factors data vectors: a) all the data rows; b) excluding the data of the minor cluster

It should be noted that the physiological value of the ULF electrocardiographic signal has been studied the least. However, there is an opinion that its power increases significantly when the body's regulatory systems are exhausted [28], [29]. Therefore, the study of the influence of the ultraslow oscillation index (ULF) on the value of the assessment of the patient's mental state based on the results of the arterial oscillogram analysis is of some interest and scientific novelty.

Figure 3 presents a two-dimensional display of the 12 ULF factors: ULF total, ULF 20, ULF 20–70, ULF 70–100, ULF 100–70, ULF 70–end, ULF per total, ULF per 20, ULF per–20–70, ULF per 70–100, ULF per 100–70, ULF per 70–end. The following parameters of UMAP have been applied: *N* neighbors = 7, minimal distance = 0.110625. Such values were determined experimentally as being optimal for subsequent classes' separation. The point colors correspond to the a priori 5 patient groups but with a numeration from 0.

According to medical research data, the ULF range considered here integrates and adapts the body's restructuring of the functional state under the influence of external factors and provides communication and coordination in the hierarchical regulation of heart rate between the cortex and lower levels that regulate the activity of the circulatory system.

Two distinct clusters are observed in Figure 3(a). The first, larger one, includes most of the patients (170 points), and the second, smaller one, consists of 11 points. The larger cluster is of greater interest because of the clearer separation of the patient groups within it. More severe cases of depression (groups 3 and 4) are situated mostly in its inner zone. The 5th group, which consists of clinical patients, is also there. The lighter depression cases (groups 1 and 2) are in the peripheral zone of the cluster. Group 3 could be considered transitional because some cases are located in the peripheral zone, while others are in the inner zone. However, all the cases in the 4th group tend to be in the inner zone. The appearance of several cases from group 3 in the peripheral zone might be explained by some subjectivity in the patients' questionnaire responses and the subsequent a priori estimation based on them.

Additionally, less clear tendencies might exist, with most of the 1st group cases (lighter ones) located further from the center than those of group 2. Similarly, points in group 3 could be more scattered than those in group 4.

The observed results suggest that the locations of points within the different zones of this major cluster align quite well with the depression levels. This is seen more clearly in Figure 3(b), where the cases of the minor cluster are excluded. The cluster is rather round and symmetrical, suggesting that the depression level might be inversely proportional to the distance of a point from the center of this conditional circle.

Nevertheless, the cases appearing in the minor cluster fall outside the established paradigm, and among them are patients from all five groups. A comparison of the mean values was conducted for the factors in the major and minor clusters, with the results presented in Table 1.

Table 1. Mean values comparison for the ULF factors in the major and minor clusters

ULF factor	The major cluster	The minor cluster
ULF total	9.0292	6.0438
ULF 20	16.2472	9.4940
ULF 20–70	28.6965	13.6575
ULF 70–100	105.2726	4777.6345
ULF 100–70	208.6256	0.0
ULF 70–end	20.9569	739.8831
ULF per total	4.1332	2.6830
ULF per 20	1.2927	1.2447
ULF per 20–70	3.3341	4.7144
ULF per 70–100	5.4696	6.3739
ULF per 100–70	5.8684	0.0
ULF per 70–end	3.7347	1.9073

From these results, it is clear that the mean values for some factors are considerably different between the cases in the major and minor clusters. Particular attention should be paid to the factors ULF 100–70 and ULF per 100–70, which have zero values for the cases in the minor cluster. Additionally, the factors ULF 70–100 and ULF 70–end have significantly greater values for the minor cluster. Evidently, it is due to such differences that the separation of the minor cluster occurred.

The nature of the observed pattern of ULF factors in the minor cluster remains somewhat unclear.

The given patients might have been affected by some unaccounted impacts, which could include medical procedures, certain diseases, special activities, and so on. In any case, when such a pattern appears – zero values for ULF 100–70 and ULF per 100–70, combined with high values for ULF 70–100 and ULF 70–end – it should be understood that these patients fall outside the paradigm and must be regarded separately. Repetition of measurements may be needed in such cases.

For subsequent analysis, the data from the minor cluster was excluded from the considered dataset.

The most obviously, two classes can be obtained from the data:

- **Class 1,** which includes the patients' groups 1 and 2, – lighter cases.
- **Class 2,** which includes the patients' groups 3, 4, and 5, – more severe cases.

For the automatic separation of the classes, the KNeighborsClassifier from the Scikit-Learn library was used. It implements the method of k-nearest neighbors. The classifier was applied to the UMAP embedding space obtained by the method described above.

The obtained accuracy values on the test data sample ranged from 0.9471 to 0.9706, depending on the randomization of the learning and test sample divisions. This represents a relatively high accuracy, especially considering that some points fall outside the accepted class separations.

Using the KNeighborsClassifier, a comparison of the main statistical characteristics was performed for these two classes, which included mean values, variation ranges, standard deviation (SD), and variation coefficient. The values of these characteristics are shown in Table 2.

Table 2. Comparison of the common statistics for the obtained classes 1 (lighter cases) and 2 (more severe cases)

ULF factor	**Characteristics values**									
	Class 1 (lighter cases)					**Class 2 (more severe cases)**				
	Mean	**Variation**		**SD**	**cov, %**	**Mean**	**Variation**		**SD**	**cov, %**
		Min.	**Max**				**Min.**	**Max**		
ULF_total	9.35	1.26	39.72	7.06	75.51	8.60	1.25	25.31	6.44	74.90
ULF_20	18.11	0.00	**460.7**	49.51	**273.3**	13.71	0.00	**73.48**	16.77	**122.4**
ULF_20–70	**37.15**	0.0	**1381.2**	141.2	**380.0**	**17.19**	0.0	**190.4**	30.11	**175.2**
ULF_70–100	**87.67**	3.60	**2829.4**	293.9	**335.3**	**129.2**	2.20	**6532.2**	766.7	**593.3**
ULF_100–70	**164.9**	0.03	**2988.4**	383.1	232.2	**268.0**	1.19	**1986.3**	463.4	172.9
ULF_70–end	21.74	0.05	**296.9**	33.51	154.2	19.90	0.04	**134.28**	26.20	131.7
ULF_per_total	3.49	0.38	12.32	2.02	57.84	5.01	0.75	16.52	3.27	65.20
ULF_per_20	1.06	0.00	3.78	0.78	73.86	1.61	0.00	5.24	1.06	65.81
ULF_per_20–70	3.44	0.0	10.33	2.49	72.48	3.19	0.0	9.14	2.62	82.09
ULF_per_70–100	5.65	0.03	12.64	2.79	49.33	5.22	0.06	13.47	2.91	55.64
ULF_per_100–70	5.53	0.00	16.02	3.54	64.01	6.33	0.04	14.73	3.74	59.17
ULF_per_70–end	3.54	0.01	**9.51**	2.72	76.97	3.99	0.01	**14.22**	3.38	84.42

As seen in Table 2, while some mean values can differ considerably between the classes, the variation intervals overlap, and the minimum values are similar for most factors. Nevertheless, it appears that factors ULF 20 and ULF 20–70 are more scattered in Class 1, while ULF 70–100 is more prominent in Class 2.

These results may have some value for specialists; however, the very low difference in minimum values of the factors and overlapping variation ranges significantly reduce the classification possibility using the raw and even normalized data. This is demonstrated by the low classification accuracy obtained on the data without UMAP transformation, where none of the classifiers in the Scikit-Learn library achieved an accuracy higher than 0.65 to 0.68 on the test sample.

7 Creation and Testing a Classifier for Depression Level Estimation on Arterial Oscillograms Data

Unfortunately, the combination of UMAP + KNeighborsClassifier (as well as any other standard classifier) has rather low predictive reliability in this context, making it unsuitable as a working tool. Despite achieving high accuracy within the UMAP embedding space (as mentioned above), the technique is not sufficiently suitable for processing new single data vectors as input data. This limitation arises due to the peculiarities of UMAP performance when using the Jacquard metric, which is tuned to operate primarily with binary data. The observed class separations occur when a relatively large 2D dataset is provided for UMAP transformation. Although the method formally allows new data sets to be input for transformation into a previously obtained embedding space, the binary nature of the metric used here may lead to unreliable results due to the lack of clearly separated clusters. As a result, division within cluster zones may not be dependable.

Nevertheless, developing a classifier capable of making reliable predictions for new data remains highly desirable. As the subsequent research has shown, this challenge could be addressed through a more detailed analysis of feature values within each class, particularly by examining correlations specific to each class.

For each of the identified classes, a correlation analysis was carried out on the ULF factors. Pairs of factors with an absolute correlation coefficient greater than 0.3 were selected. Among these, pairs were chosen based on the significance of their correlations within a specific class or if the correlation coefficients had differing signs across classes (though this latter case was not observed here). For Class 1, the following factor pairs exhibit significant correlations unique to this class (but not to Class 2):

ULF_total and ULF_per_20	$r = 0.4506$
ULF_20 and ULF_per_20	$r = 0.3712$
ULF_70–end and ULF_per_70–end	$r = 0.5243$

For the Class 2 those are as follows:

ULF_total and ULF_70–end	$r = 0.4573$
ULF_total and ULF_per_total	$r = –0.5731$
ULF_20 and ULF_70–end	$r = 0.3899$
ULF_20 and ULF_per_total	$r = –0.4465$
ULF_20–70 and ULF_100–70	$r = 0.3035$
ULF_20–70 and ULF_per_20	$r = –0.3076$
ULF_100–70 and ULF_per_total	$r = –0.3269$
ULF_70–end and ULF_per_total	$r = –0.3594$
ULF_70–end and ULF_per_100–70	$r = –0.3341$
ULF_per_20 and ULF_per_20–70	$r = –0.3311$
ULF_per_20–70 and ULF_per_70–100	$r = 0.3193$
ULF_per_20–70 and ULF_per_100–70	$r = 0.3896$

As we can see, Class 2 contains significantly more pairs of correlated ULF factors than those specific to Class 1. No feature pairs with differing correlation coefficient signs were observed between Class 1 and Class 2.

Additionally, it was found that ULF 70–100 factors have rather weak correlations with all the other ULF features in both Class 1 and Class 2.

It has been suggested that the classifier should use not only the values themselves but also the products of features that show significant correlations specific to certain classes. The influence of adding and removing such features on classifier accuracy was studied. Combinations that increased accuracy were selected, while those that tended to decrease accuracy were removed.

As a result, the following list of classification parameters was formed:

Separate features:

ULF_70–100
ULF_per_70–100

Products of features:

ULF_20 * ULF_per_20
ULF_total * ULF_70–end
ULF_total * ULF_per_total
ULF_20 * ULF_70–end
ULF_20–70 * ULF_100–70
ULF_20–70 * ULF_per_20
ULF_100–70 * ULF_per_total
ULF_70–end * ULF_per_total
ULF_70–end * ULF_per_100–70
ULF_per_total * ULF_per_20–70
ULF_per_20–70 * ULF_per_70–100
ULF_per_20–70 * ULF_per_100–70

The given set of features increased the classifier's accuracy on the test data to 0.7785, which is a significant improvement; however, for practical usage, a higher accuracy is desirable. Therefore, it was suggested that more parameters be added to the model, specifically products of the ULF factors with other features. To this end, a series of additional experiments were carried out. Sequentially, products of the ULF factors with each of the other features were added to the model one by one. The combination that increased the model's accuracy the most was selected. The experiment was then repeated with the newly supplemented model.

As a result, the following feature products were added to the model:

ULF_per_total * systola_2
ULF_per_70–end * O_IVR_neg
ULF_70–100 * O_L2_pos
ULF_70–100 * Hurst_20–70
ULF_70–100 * O_Mo_neg
ULF_70–end * V
ULF_per_20–70 * HFx18x21_20
ULF_per_total * HF_70–100_int_p
ULF_per_20–70 * S_Hil_25_60Hz_total
ULF_per_20–70 * S_Hil_HFx13x15_20
ULF_per_20–70 * S_Hil_HFx15x18_20

The suggested supplementation increased the model's accuracy to between 0.9515 and 0.9792 on the test data. The best accuracy values were achieved with the KNeighborsClassifier, LGBMClassifier, and AdaBoostClassifier, while the RandomForestClassifier and ExtraTreesClassifier also performed adequately for this purpose. The obtained accuracy appears suitable for a production classification service that could be developed using the proposed approach.

Additional experiments were conducted to evaluate the possibility of detecting more classes using the suggested input feature set. The results are shown in Table 3.

From Table 3, it is evident that attempts to separate classes differently generally result in lower precision, with the exception of isolating Class 5 from all others, which occurs with nearly 100%

probability. Moreover, the a priori classes 1, 2, and 3 can be reasonably distinguished from 4 and 5. In this case, a certain asymmetry in the blurring of the intermediate Class 3 may play a role. At the same time, Classes 4 and 5 can also be separated without loss of accuracy. However, there is no reliable automatic division into all 5 a priori classes. Additionally, the separation of Classes 1 and 2 is not reliable, as indicated by the decrease in model accuracy when attempting such a division. Perhaps the selection of alternative initial factors would allow for a more dependable separation of these classes as well.

Table 3. Results of classification experiments with different groupings of a priori classes

Classes grouping	The highest accuracy value	Classifier
5 a priori classes	0.5844	BaggingClassifier
1 and 2 to be Class 1 3, 4, 5 to be Class 2	0.9519	KNeighborsClassifier
1 to be Class 1 2 to be Class 2 3, 4, 5 to be Class 3	0.6887	KNeighborsClassifier
1 and 2 – to be Class 1 3 to be Class 2 4 and 5 to be Class 3	0.7974	KNeighborsClassifier
1, 2, 3 – Class 1 4 and 5 – Class 2	0.8826	XGBClassifier, KNeighborsClassifier
1, 2, 3 to be Class 1 4 to be Class 2 5 to be Class 3	0.8867	XGBClassifier
1 and 2 to be Class 1 3 and 4 to be Class 2 5 to be Class 3	0.8543	LGBMClassifier
1, 2, 3, 4 to be Class 1 5 to be Class 2	0.9990	XGBClassifier
1 to be Class 1 2, 3, 4 to be Class 2 5 to be Class 3	0.7465	LGBMClassifier

For practical tasks, the simultaneous application of two classifiers is possible. The first classifier, the KNeighborsClassifier, should be trained to distinguish between Classes 1 and 2 (less complex cases) and Classes 3, 4, and 5 (more complex cases). The second classifier, the XGBClassifier, should be trained to separate Class 5 (clinical cases) from all others. Thus, a more reliable classification into three classes can be obtained: Class 1 – consisting of cases 1 and 2 (less complex cases), Class 2 – comprising cases 3 and 4 (more complex cases), and Class 3 – representing Class 5 (clinical cases).

It is worth noting that the approach proposed here, along with the associated software modules, has the potential to be used not only within the domain discussed here but also for other purposes in medicine and other practical diagnostic areas. The reuse of software development artifacts, such as the modules of the proposed classification system, is vital for improving the efficiency and effectiveness of software lifecycle processes. Adopting reusable components not only accelerates development but also ensures a higher degree of consistency and reliability across different phases of the software lifecycle. When integrated with a domain engineering approach, which focuses on reusing artifacts within a specific domain, this strategy becomes even more powerful. Domain engineering, combined with semantic analysis, allows for a more systematic reuse of modules by ensuring they are adapted and optimized for specific contexts and tasks [34]–[37].

8 Conclusion

Clustering of arterial oscillogram data for a sample of patients was carried out based on ULF indicators (power of ultra-low-frequency oscillations with a frequency of less than 0.003 Hz) using

UMAP for dimensionality reduction. This analysis revealed a potential relationship between these factors and depression levels. Two distinct classes were objectively identified, which can be conventionally characterized as lighter cases and more severe cases. This finding facilitated the subsequent development of an automated classifier using machine learning methods.

During classifier development, it was found that the direct use of ULF indicators alone resulted in relatively low classification accuracy on the test sample. However, it was established that each class exhibited unique correlations among ULF indicators. Therefore, only a few ULF parameters were directly used as input features for classification, while other parameters were incorporated as products of correlated features. This approach improved classification accuracy to 0.7785 using the nearest neighbors method.

To further enhance accuracy, additional model parameters were introduced by combining ULF indicators with other measured factors, resulting in an accuracy increase to 0.9515–0.9792, depending on the training and test sample division, also using the k-nearest neighbors method. Additionally, it was found that using the XGBClassifier, an additional class representing clinical cases could be identified with nearly 100% accuracy. While other classification approaches yielded lower accuracy, a considerably reliable (0.88) separation of a fourth class (encompassing the most severe and clinical cases) was also achievable with a separately trained XGBClassifier.

These results indicate the feasibility of building an automated diagnostic AI service that can reliably estimate at least four levels of depression based on arterial oscillogram analysis. To implement such a classifier, at least three trained models are needed: one to distinguish between lighter and more severe cases using KNeighborsClassifier, another to separate severe cases from clinical cases using XGBClassifier, and a third to isolate clinical cases from all others using XGBClassifier.

The main scientific contributions of this research in terms of complex systems analysis and classifier development include:

1 Extracting a base feature set from a general dataset where the number of features significantly exceeds the number of data rows, and using clustering in reduced dimensional space to identify objectively separable classes.

2 Accounting for correlations between features that are specific to one or a few classes, and using products of these features as input parameters to enhance classification accuracy.

3 Developing a methodology for expanding the base feature set by searching for class-specific correlations with additional features, and selecting those which most significantly improve classification accuracy.

Acknowledgements

This study would not have been possible without the financial support of the National Research Foundation of Ukraine (Open Funder Registry: 10.13039/100018227). Our work was funded by Grant contract: Development of the cloud-based platform for patient-centered telerehabilitation of oncology patients with mathematical-related modeling, application ID: 2021.01/0136.

We acknowledge also Petro Stetsyk and Oleksandr Palagin who became, the winners of the competition “Science for Security and Sustainable Development of Ukraine,” conducted by the National Research Foundation of Ukraine, contributing tour research

In memoriam, we acknowledge university teacher Yuriy Tikhonov, who profoundly influenced the academic journey of his students with his deep knowledge and experience.

Data Availability Statement

The data supporting this study’s findings are available from the corresponding author, Kyrylo Malakhov, upon eligible request.

References

[1] C. G. Caro, T. J. Pedley, R. C. Schroter, W. A. Seed, and K. H. Parker, *The Mechanics of the Circulation*, 2nd ed., Cambridge University Press, 2011. Available: https://doi.org/10.1017/CBO9781139013406

[2] D. Vakulenko, L. Vakulenko, L. Hryshchuk, and L. Sas, "Application Arterial Oscilography to Study the Adaptive Capacity of Subject with COVID-19 in Primary Care," in *Primary Health Care*, A. Emel Önal, Ed., IntechOpen, 2022. Available: https://doi.org/10.5772/intechopen.98570

[3] V. P. Martsenyuk, D. V. Vakulenko, L. A. Hryshchuk, L. O. Vakulenko, N. O. Kravets, and N. Ya. Klymuk, "On the Development of Directed Acyclic Graphs in Differential Diagnostics of Pulmonary Diseases with the Help of Arterial Oscillogram Assessment," in *Graph-Based Modelling in Science, Technology and Art*, S. Zawiślak and J. Rysiński, Eds., in Mechanisms and Machine Science, vol. 107, Springer, 2022, pp. 157–173. Available: https://doi.org/10.1007/978-3-030-76787-7_8

[4] A. Pokrovsky, *Clinical Angiology*. Medicine, 1979 (in Russian). Available: https://cdn.e-rehab.pp.ua/u/pokrovsky-clinical-angiology-1979.pdf

[5] H. R. Warner, S. H. Swan, and D. C. Connolly, "Quantitation of beat-to-beat changes in stroke volume from the aortic pulse contour in man," *J Appl Physiol.*, vol. 5, no. 9, pp. 495–507, 1953. Available: https://doi.org/10.1152/jappl.1953.5.9.495

[6] G. J. Langewouters, K. H. Wesseling, and W. J. Goedhard, "The static elastic properties of 45 human thoracic and 20 abdominal aortas in vitro and the parameters of a new model," *J Biomech*, vol. 17, no. 6, pp. 425–435, 1984. Available: https://doi.org/10.1016/0021-9290(84)90034-4

[7] O. V. Palagin, V. Yu. Velychko, K. S. Malakhov, and O. S. Shchurov, "Research and development workstation environment: The new class of current research information systems," in *Proceedings of the 11th International Conference of Programming UkrPROG 2018*, CEUR Workshop Proceedings, CEUR-WS, vol. 2139, 2018, pp. 255–269. Available: http://ceur-ws.org/Vol-2139/255-269.pdf

[8] O. V. Palagin, V. V. Kaverinskiy, A. Litvin, and K. S. Malakhov, "OntoChatGPT Information System: Ontology-Driven Structured Prompts for ChatGPT Meta-Learning," *Int. J. Comput.*, vol. 22, no. 2, pp. 170–183, 2023. Available: https://doi.org/10.47839/ijc.22.2.3086

[9] O. V. Palagin, V. V. Kaverinskiy, K. S. Malakhov, and M. G. Petrenko, "Fundamentals of the Integrated Use of Neural Network and Ontolinguistic Paradigms: A Comprehensive Approach," *Cybern. Syst. Anal.*, vol. 60, no. 1, pp. 111–123, 2024. Available: https://doi.org/10.1007/s10559-024-00652-z

[10] M. G. Petrenko, E. Cohn, O. Shchurov, and K. S. Malakhov, "Ontology-Driven Computer Systems: Elementary Senses in Domain Knowledge Processing," *South Afr. Comput. J.*, vol. 35, no. 2, pp. 127–144, 2023. Available: https://doi.org/10.18489/sacj.v35i2.17445

[11] K. S. Malakhov, M. G. Petrenko, and E. Cohn, "Developing an ontology-based system for semantic processing of scientific digital libraries," *South Afr. Comput. J.*, vol. 35, no. 1, pp. 19–36, 2023. Available: https://doi.org/10.18489/sacj.v35i1.1219

[12] O. V. Palagin, V. Yu. Velychko, K. S. Malakhov, and O. S. Shchurov, "Distributional semantic modeling: A revised technique to train term/word vector space models applying the ontology-related approach," in *Proceedings of the 12th International Scientific and Practical Conference of Programming (UkrPROG 2020)*, CEUR Workshop Proceedings, CEUR-WS, vol. 2866, 2020, pp. 342–353. Available: http://ceur-ws.org/Vol-2866/ceur_342-352palagin34.pdf

[13] The Government of Ukraine, "National Research Foundation of Ukraine," *NRFU*, 2023. Available: https://nrfu.org.ua

[14] K. S. Malakhov, "Letter to the Editor – Update from Ukraine: Development of the Cloud-based Platform for Patient-centered Telerehabilitation of Oncology Patients with Mathematical-related Modeling," *Int. J. Telerehabilitation*, vol. 15, no. 1, pp. 1–3, 2023. Available: https://doi.org/10.5195/ijt.2023.6562

[15] K. S. Malakhov, "Innovative Hybrid Cloud Solutions for Physical Medicine and Telerehabilitation Research," *Int. J. Telerehabilitation*, vol. 16, no. 1, pp. 1–19, 2024. Available: https://doi.org/10.5195/ijt.2024.6635

[16] K. S. Malakhov, S. V. Kotlyk, and M. G. Petrenko, "Theoretical Aspects of Transdisciplinarity in Telerehabilitation," *Int. J. Telerehabilitation*, no. Special Issue: Research Status Report – Ukraine in 2024, pp. 1–13, 2024. Available: https://doi.org/10.5195/ijt.2024.6643

[17] O. V. Palagin, K. S. Malakhov, V. Yu. Velychko, T. V. Semykopna, and O. S. Shchurov, "Hospital Information Smart-System for Hybrid E-Rehabilitation," in *Proceedings of the 13th International Scientific and Practical*

Programming Conference (UkrPROG 2022), CEUR Workshop Proceedings, CEUR-WS, vol. 3501, 2022, pp. 140–157. Available: https://ceur-ws.org/Vol-3501/s50.pdf

[18] L. McInnes, J. Healy, N. Saul, and L. Großberger, “UMAP: Uniform Manifold Approximation and Projection,” *J. Open Source Softw.*, vol. 3, no. 29, p. 861, 2018. Available: https://doi.org/10.21105/joss.00861

[19] T. Cover and P. Hart, “Nearest neighbor pattern classification,” *IEEE Trans. Inf. Theory*, vol. 13, no. 1, pp. 21–27, 1967. Available: https://doi.org/10.1109/TIT.1967.1053964

[20] N. S. Altman, “An Introduction to Kernel and Nearest-Neighbor Nonparametric Regression,” *Am. Stat.*, vol. 46, no. 3, pp. 175–185, 1992. Available: https://doi.org/10.1080/00031305.1992.10475879

[21] T. Chen and C. Guestrin, “XGBoost: A Scalable Tree Boosting System,” in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, San Francisco California USA: ACM, Aug. 2016, pp. 785–794. Available: https://doi.org/10.1145/2939672.2939785

[22] G. Ke *et al.*, “LightGBM: A Highly Efficient Gradient Boosting Decision Tree,” in *Advances in Neural Information Processing Systems*, pp. 3146–3154, 2017. Available: https://proceedings.neurips.cc/paper_files/paper/2017/hash/6449f44a102fde848669bdd9eb6b76fa-Abstract.html. Accessed on Oct. 17, 2024.

[23] Y. Haque *et al.*, “State-of-the-Art of Stress Prediction from Heart Rate Variability Using Artificial Intelligence,” *Cogn. Comput.*, vol. 16, no. 2, pp. 455–481, 2024. Available: https://doi.org/10.1007/s12559-023-10200-0

[24] D. Geng, Q. An, Z. Fu, C. Wang, and H. An, “Identification of major depression patients using machine learning models based on heart rate variability during sleep stages for pre-hospital screening,” *Comput. Biol. Med.*, vol. 162, p. 107060, 2023. Available: https://doi.org/10.1016/j.compbiomed.2023.107060

[25] B. Xia, N. Innab, V. Kandasamy, A. Ahmadian, and M. Ferrara, “Intelligent cardiovascular disease diagnosis using deep learning enhanced neural network with ant colony optimization,” *Sci. Rep.*, vol. 14, no. 1, p. 21777, 2024. Available: https://doi.org/10.1038/s41598-024-71932-z

[26] Z. Yang, C. Chen, H. Li, L. Yao, and X. Zhao, “Unsupervised Classifications of Depression Levels Based on Machine Learning Algorithms Perform Well as Compared to Traditional Norm-Based Classifications,” *Front. Psychiatry*, vol. 11, p. 45, 2020. Available: https://doi.org/10.3389/fpsyt.2020.00045

[27] D. V. Vakulenko and L. O. Vakulenko, *Arterial Oscillography: New Capabilities of the Blood Pressure Monitor with the Oranta-AO Information System*. in New Developments in Medical Research. Nova Science Publishers, 2024. Available: https://doi.org/10.52305/XFFR7057

[28] P. M. Bayevsky and G. G. Ivanov, “Cardiac Rhythm Variability: The Theoretical Aspects and the Opportunities of Clinical Application,” *Ultrasound Funct. Diagn.*, vol. 2001, no. 3, pp. 106–127, 2001 (in Russian). Available: http://vidar.ru/article.asp?fid=USFD_2001_3_108

[29] M. Malik, J. Camm, T. Bigger, and S. Cerutti, “Heart Rate Variability: Standards of Measurement, Physiological Interpretation, and Clinical Use,” *Circulation*, vol. 93, no. 5, pp. 1043–1065, 1996. Available: https://doi.org/10.1161/01.CIR.93.5.1043

[30] Y. A. LeCun, L. Bottou, G. B. Orr, and K.-R. Müller, “Efficient BackProp,” in *Neural Networks: Tricks of the Trade*, *Lecture Notes in Computer Science*, vol. 7700, Springer, 2012, pp. 9–48. Available: https://doi.org/10.1007/978-3-642-35289-8_3

[31] I. T. Jolliffe and J. Cadima, “Principal component analysis: a review and recent developments,” *Philos. Trans. R. Soc. Math. Phys. Eng. Sci.*, vol. 374, no. 2065, p. 20150202, 2016. Available: https://doi.org/10.1098/rsta.2015.0202

[32] L. McInnes and J. Healy, “UMAP: Uniform Manifold Approximation and Projection for Dimension Reduction,” *ArXiv e-prints 1802.03426*, 2018. Available: https://umap-learn.readthedocs.io/en/latest/

[33] I. Guyon, J. Weston, S. Barnhill, and V. Vapnik, “Gene Selection for Cancer Classification using Support Vector Machines,” *Mach. Learn.*, vol. 46, no. 1, pp. 389–422, 2002. Available: https://doi.org/10.1023/A:1012487302797

[34] O. Chebanyuk, “Investigation of Drawbacks of the Software Development Artifacts Reuse Approaches based on Semantic Analysis,” in *Advances in Computer Science for Engineering and Education VI*, *Lecture Notes on Data Engineering and Communications Technologies*, vol. 181, Springer, 2023, pp. 514–523. Available: https://doi.org/10.1007/978-3-031-36118-0_46

[35] O. Chebanyuk, “Software Reuse Approach Based on Review and Analysis of Reuse Risks from Projects Uploaded to GitHub,” in *Computer Science and Education in Computer Science*, *Lecture Notes of the Institute*

for Computer Sciences, Social Informatics and Telecommunications Engineering, vol. 514, Springer, 2023, pp. 144–155. Available: https://doi.org/10.1007/978-3-031-44668-9_11

[36] V. Opanasenko, A. Palahin, and S. Zavyalov, “The FPGA-Based Problem-Oriented On-Board Processor,” in *2019 10th IEEE International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications (IDAACS)*, IEEE, 2019, pp. 152–157. Available: https://doi.org/10.1109/IDAACS.2019.8924360

[37] V. M. Opanasenko, Sh. Kh. Fazilov, S. S. Radjabov, and Sh. S. Kakharov, “Multilevel Face Recognition System,” *Cybern. Syst. Anal.*, vol. 60, no. 1, pp. 146–151, 2024. Available: https://doi.org/10.1007/s10559-024-00655-w