



# Article SGD-Based Cascade Scheme for Higher Degrees Wiener Polynomial Approximation of Large Biomedical Datasets

Ivan Izonin <sup>1,\*</sup>, Roman Tkachenko <sup>2</sup>, Rostyslav Holoven <sup>3</sup>, Kyrylo Yemets <sup>1</sup>, Myroslav Havryliuk <sup>1</sup> and Shishir Kumar Shandilya <sup>4</sup>

- <sup>1</sup> Department of Artificial Intelligence, Lviv Polytechnic National University, 79013 Lviv, Ukraine
- <sup>2</sup> Department of Publishing Information Technologies, Lviv Polytechnic National University, 79013 Lviv, Ukraine
- <sup>3</sup> Department of System Design, Ivan Franko National University of Lviv, 79005 Lviv, Ukraine
- <sup>4</sup> School of Computing Science and Engineering, VIT Bhopal University, Bhopal 466114, India
- \* Correspondence: ivan.v.izonin@lpnu.ua

Abstract: The modern development of the biomedical engineering area is accompanied by the availability of large volumes of data with a non-linear response surface. The effective analysis of such data requires the development of new, more productive machine learning methods. This paper proposes a cascade ensemble that combines the advantages of using a high-order Wiener polynomial and Stochastic Gradient Descent algorithm while eliminating their disadvantages to ensure a high accuracy of the approximation of such data with a satisfactory training time. The work presents flow charts of the learning algorithms and the application of the developed ensemble scheme, and all the steps are described in detail. The simulation was carried out based on a real-world dataset. Procedures for the proposed model tuning have been performed. The high accuracy of the approximation based on the developed ensemble scheme was established experimentally. The possibility of an implicit approximation by high orders of the Wiener polynomial with a slight increase in the number of its members is shown. It ensures a low training time for the proposed method during the analysis of large datasets, which provides the possibility of its practical use in the biomedical engineering area.

**Keywords:** ensemble learning; biomedical engineering; approximation; Wiener polynomial; SGD; fast machine learning; cascade ensemble

# 1. Introduction

Biomedical engineering as a science was formed in the 1950s. As an interdisciplinary field of knowledge, it combines engineering and medical knowledge to solve various complex problems [1]. The development of smart medical equipment and microelectromechanical systems, the development of clinical engineering and bioinformatics, and many other specializations of biomedical engineering rely on intelligent data analysis. It is facilitated by the rapid modern growth of computing power, the appearance of various portable devices for collecting information, broadband Internet access, etc. [2,3]. All this provides a foundation for building smart systems that will combine technical and medical–biological knowledge to increase the efficiency of decision-making processes. In addition, the modern development of most specializations and areas of research in biomedical engineering is characterized by the collection of a huge amount of information. These are tabular datasets, images, videos, biosignals, etc. [4–6]. All this requires effective methods for the intellectual analysis of such data.

The need to process tabular datasets is characteristic of most biomedical engineering specializations. This is actually the collection of tabular data or the transformation of signals or images into tabular sets in the form of extracted features, etc. [7,8]. That is why the improvement of the best existing techniques, as well as the development of new models and methods for the intelligent analysis of tabular datasets, is an urgent task today. It is not



Citation: Izonin, I.; Tkachenko, R.; Holoven, R.; Yemets, K.; Havryliuk, M.; Shandilya, S.K. SGD-Based Cascade Scheme for Higher Degrees Wiener Polynomial Approximation of Large Biomedical Datasets. *Mach. Learn. Knowl. Extr.* 2022, *4*, 1088–1106. https://doi.org/10.3390/make4040055

Academic Editor: Krzysztof J Cios

Received: 14 October 2022 Accepted: 17 November 2022 Published: 21 November 2022

**Publisher's Note:** MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



**Copyright:** © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https:// creativecommons.org/licenses/by/ 4.0/). straightforward, both by the large volumes of collected data and the high dimensionality of such data. Important tasks that, ideally, should be solved simultaneously when using machine learning methods, in this case, are the following:

- Ensuring the highest possible approximation/classification accuracy via the selected method of intellectual analysis;
- Providing high generalization properties of the model based on such an analysis;
- Guaranteeing the high speed of the intelligent analysis method, particularly in the training mode.

The ability to build effective software and hardware smart systems for medical use depends largely on the effective solution to these three problems [9]. This will significantly affect the possibility or effectiveness of their practical application when solving real-world problems in various specializations and areas of biomedical engineering research [10,11].

The existing machine learning methods from the linear class are often used for solving applied problems of biomedical engineering [12] as they provide the highest speed of operation. However, such a models' approximation/classification accuracy and generalization properties are lost in this case. An example would be research [13–16], which demonstrates the high speed of linear methods, but the low accuracy of their work.

The machine learning methods from the class of non-linear, on the contrary, require more time to implement training procedures [17–19]. However, on the other hand, they can increase the prediction accuracy of models embedded in their basis [20]. An example of applying such methods in the biomedical engineering area is solving the material classification tasks in the production of a medical implant [21].

Ensemble learning methods, which have gained a significant popularity in recent years, provide a high prediction/classification accuracy and increase the generalization properties compared to the single machine learning methods [22–24]. Despite this, some methods in this area require a lot of computing resources and memory for their practical implementation in biomedical engineering tasks [25,26]. It is also reflected in the duration of their training procedures.

Let us consider in more detail the three primary classes of building ensemble methods (Figure 1): bootstrap aggregating, boosting, and stacking.



Figure 1. Three main classes of ensemble methods.

Bootstrap aggregating or bagging (Figure 1 left) is based on using two main steps: bootstrap and aggregation. The idea of the method is to divide a large sample of data into smaller ones that do not correlate with each other (bootstrap) and to process them in parallel [27]. The final result of the model is formed based on the generalization of the results of all models (aggregation). The disadvantage of this approach is that each model will process only a part of the entire dataset, which should be representative and, at the same time, not correlated with others. In addition, it is necessary to carefully choose the averaging method for the regression task or the voting method for the classification task, which will best combine the solutions obtained from all ensemble elements. An example of applying such methods in a biomedical data analysis is [28].

Boosting is based on training the model iteratively so that the current model's training depends on the previous models' results [29]. That is, learning of this class of methods will take place only sequentially. It should be noted that each subsequent model focuses on processing the data that the previous one could not handle. Such a coherent adaptation of the weak predictors ensures the construction of one strong predictor. It is due to a step-by-step increase in the prediction accuracy while analyzing the most complex sample objects obtained from previous models. Despite this, the main disadvantages of this ensemble model are that it is sensitive to outliers and it is almost impossible to scale up. The prediction of the medical treatment in patients with acute bronchiolitis using such an approach is described in [30].

The idea of methods from the third ensemble strategy, stacking, is to train several weak models (which can be different machine learning methods) and combine them to train a metamodel that will generalize the prediction [31]. This strategy is quite interesting due to the possibility of a parallel processing of the different machine learning methods and implementing the second learning step: a meta-algorithm to increase the prediction/classification accuracy. However, none of the weak stacking predictors uses the entire dataset for analysis. In addition, such a strategy requires considerable resources to train each ensemble element. Moreover, the practical application of stacking involves the selection of optimal parameters for each ensemble member, which will be completely different. Implementing such procedures when analyzing large datasets takes a lot of time. The promising stacking-based approach applied to solve the mortality risk prediction of COVID-19 patients is presented in [32].

In general, ensemble methods can reduce variance using the bagging strategy, reduce bias using the boosting methods, or improve the prediction accuracy using the stacking approach. However, considering the considerable resources required for operation methods from the above three classes, their application is limited to small tasks during the analysis of large volumes of data.

Processing large datasets in the biomedical engineering area should be simple and accurate. Based on these considerations, the authors of [33] developed a method for the combined use of the quadratic Wiener polynomial and SGD to improve the accuracy and speed of the data approximation. Such a combination eliminates the disadvantages of both methods, bringing only their advantages into the combined model. In particular, the accuracy of an SGD operation is improved due to the high approximation properties of SGD. On the other hand, the search for Wiener polynomial coefficients is significantly accelerated using SGD. However, the proposed scheme will not be effective when analyzing large datasets with a significant nonlinearity. In this case, it is necessary to use a high orders Wiener polynomial to approximate significantly non-linear response surfaces. However, this approach will lead to an almost unrealistic growth of its members, which will reduce the accuracy and generalization properties of SGD during their analysis. In addition, this approach can provoke an overfitting. Moreover, in the case of vast amounts of data, it will require a lot of time and resources to implement training, even for the SGD algorithm.

Therefore, this paper aims to design a new cascade-based ensemble scheme of a high-degree Wiener polynomial approximation using the SGD algorithm to improve the performance of solving prediction tasks in biomedical engineering for cases of large dataset processing.

The main contribution of this paper can be summarized as follows:

- We designed a new ensemble scheme for a higher degree's Wiener polynomial approximation using SGD regressors that provide a high performance during the analysis of large datasets in the biomedical engineering area;
- We chose the optimal parameters of the designed ensemble (loss of the function of the SGD algorithm, Wiener polynomial degree, and cascade levels that help us to obtain a higher prediction accuracy with strong generalization properties and decrease the duration of its training time;
- We show a higher prediction accuracy and speed of the proposed ensemble scheme when solving the heart rate prediction task using large datasets compared with the existing methods.

The structure of the paper is as follows: prerequisites and details of the proposed ensemble model are described in Section 2. Section 3 contains the results of the modeling and optimal parameters selections procedures. A comparison and discussion are presented in Section 4. Section 5 contains the conclusions and prospects for future research.

#### 2. Materials and Methods

Many biomedical engineering tasks are characterized by large volumes of data intended for analysis. Machine learning methods are used for their effective processing. However, they do not always provide a sufficient approximation accuracy, especially in the case of complex non-linear response surfaces. In this case, we can apply a non-linear expansion of the inputs to increase the accuracy of its analysis. One of the options for implementing such an approach is the use of a quadratic Wiener polynomial. However, in the case of very complex response surfaces, the quadratic polynomial approximation does not provide a sufficient accuracy. In these cases, it is worth using higher orders of this polynomial. However, during the analysis of large volumes of data, this approach is accompanied by a significant increase in the training time, and in the case of using polynomial orders higher than 3, a significant complication of the training procedure.

This paper proposes a new ensemble scheme for approximation by the Wiener polynomial of high orders in an implicit form. It is characterized by a significantly lower complexity of the training procedure compared to the use of direct approximation by high orders of this polynomial.

The advanced ensemble method is based on the principles of cascading machine learning methods and the use of SGD for the high-speed identification procedure of its members. Let us consider in detail all the components of the proposed approach in more detail.

# 2.1. Wiener Polynomial

As a discrete analog of the Volterra series, the Wiener polynomial is often used to solve problems of the approximation of non-linear dependencies [34,35]. In particular, it is the basis of the well-known group method of data handling [36]. However, in this case, the quadratic Wiener polynomial is usually used. It provides a sufficient prediction accuracy in cases of the analysis of medium-sized datasets with a small level of nonlinearity [37]. In this case, the search for its coefficients is carried out using the least squares method [38]. The general form of this polynomial can be represented as follows [33]:

$$Y(x_1,...,x_n) = \beta_i + \sum_{i=1}^n \beta_i x_i + \sum_{i=1}^n \sum_{j=i}^n \beta_{i,j} x_i x_j + \sum_{i=1}^n \sum_{j=i}^n \sum_{l=j}^n \beta_{i,j,l} x_i x_j x_l + \ldots + \sum_{i=1}^n \sum_{j=i}^n \sum_{l=j}^n \ldots \sum_{z=k-1}^n \beta_{i,j,l,...,z} x_i x_j x_l \ldots x_z,$$

where  $\beta_i$  are the polynomial coefficients that should be found by the chosen method;  $x_1, \ldots, x_n$  are the inputs attributes and Y is a searching parameter that should be predicted.

The main drawback of using the quadratic Wiener polynomial is that it does not provide a satisfactory approximation accuracy in the case of very complex, non-linear response surfaces [39], which are characteristic of many applied biomedical engineering

tasks. Additionally, in this case, the least squares method is not the best option for finding coefficients for its members [40].

To eliminate this shortcoming in the case of processing medium and large datasets, in [33], the authors proposed the use of SGD. Let us consider its work in more detail.

#### 2.2. SGD

The class of gradient methods includes many optimization algorithms that are used in machine learning. In particular, a classical gradient descent is used to find the minimum value of the loss function. That is, obtaining the smallest possible error and increasing the prediction/classification accuracy. It should be noted here that the used loss functions may be different. Detailed mathematical explanations of the work of this method are given in [41].

Even though the gradient descent is an iterative method where the gradient vector of the objective function is considered at each step, it is characterized by the simplicity of its implementation. We considered two main options for implementing a gradient descent: batch and stochastic. In the first case, each iteration of the algorithm involves processing the entire training sample, and only after that are the weighting coefficients adjusted. In this case, the gradient is calculated over the entire available training sample. This approach can be computationally complex and therefore can only be effective when processing short and medium-sized datasets [42]. In the case of processing large volumes of data, it is not optimal. The stochastic version of the gradient descent eliminates this drawback [43]. In this case, only one subsample from N is randomly selected at each algorithm iteration. That is, updating the weighting coefficients takes place only based on the processing of this random subsample.

Among the disadvantages of this approach, the use of approximate gradients should be noted, which leads to a general approximate estimate of the loss function. However, the main advantage of SGD is the high speed of the learning process on extensive data. This advantage became the main argument for using SGD in the developed ensemble scheme since the volume of the input data is sufficiently large.

This paper uses a variant of the non-linear expansion of the inputs based on the Wiener polynomial to improve the accuracy of an SGD operation. Using the Wiener polynomial and SGD provides a significantly higher approximation accuracy with a significant reduction in the time of obtaining the coefficients of the Wiener polynomial members compared to the least squares method. However, in the case of the need to approximate by this polynomial with higher orders, the dimension of the input data space and the SGD training time will increase significantly [33]. Accordingly, this approach will not be optimal for analyzing large datasets. In this paper, we developed a new ensemble scheme to eliminate the shortcomings mentioned above.

#### 2.3. Proposed Ensemble Scheme Using Wiener Polynomial and SGD

The ensemble scheme developed in this work is intended for processing large datasets. It is based on the method of the approximation of response surfaces based on using the Wiener polynomial and SGD, which was developed in [33]. The authors of [33] show an increase in the approximation accuracy using higher orders of this polynomial. However, in analyzing large sets of biomedical data, the developed approach will be very resource and time consuming. In addition, a significant increase in the number of independent features, the characteristic of high orders of the Wiener polynomial, can provoke an overfitting.

In order to avoid all the shortcomings mentioned above of the existing method, the developed ensemble scheme is based on an approach to cascade the machine learning methods. In this case, the number of independent features of an input dataset using the quadratic Wiener polynomial does not increase significantly. The use of several levels of the ensemble ensures the reduction in its errors. Using one of the fastest machine learning methods, SGD, ensures a high performance, especially when analyzing large sets of biomedical data.

In more detail, let us consider the training and application algorithms of the developed ensemble scheme.

2.3.1. Training Algorithm for the Proposed Scheme

The available dataset is divided into the training and test samples to implement both the training and application algorithms of the developed approach. Both sets are normalized. In this paper, we used the Min–Max scaler.

To implement the training procedure, the training sample must be divided into parts (datasample1, datasample2, etc., datasampleN). Each part will correspond to each new node from *N* nodes of the cascade scheme. At each node, the inputs will be expanded nonlinearly using the Wiener polynomial, and its coefficients will be searched based on the SGD algorithm. A feature of the developed scheme is that each subsequent node of the cascade will process its data sample, containing an additional attribute: the output from the previous node of the developed scheme.

Figure 2 shows a flowchart of the training algorithm of the proposed ensemble scheme using the Wiener polynomial and SGD.



**Figure 2.** Architecture of the proposed ensemble scheme using Wiener polynomial and SGD: (training mode).

Therefore, the algorithmic implementation of the training procedure for the developed ensemble scheme will consist of the following steps:

- 1. We perform a non-linear expansion of the inputs for *datasample1* based on (1). Then, we train the SGD of the first node of the ensemble (*SGD\_1*);
- 2. We apply datasample2 on the previously trained node (*SGD\_1*) from step 1. We add the predicted output as a new independent feature to *datasample2*. We perform procedure (1) and train the SGD of the second node of the ensemble (*SGD\_2*);
- 3. We perform steps 1 and 2 for *datasample3* in application mode. We operate (1) on *datasample3* extended by one independent variable as a result of step 2, and train the SGD of the third node of the ensemble (*SGD\_3*);

4. .....

5. We sequentially perform all the previous steps in the application mode to train the last of the *N* nodes of the ensemble. Next, we apply (1) to the expanded *datasample3* and perform the SGD training procedure of the last node of the ensemble (*SGD\_N*).

As a result of performing all the above actions, we get a pre-trained cascade ensemble of *N*-nodes, where *N* determines the number of data samples into which the training data sample was divided.

# 2.3.2. An Application Algorithm for the Proposed Scheme

The application mode is characterized by having a dataset or one data vector with an unknown output to be predicted, as well as a pre-trained cascade ensemble with *N* nodes.

The algorithmic implementation of the procedure for applying the developed ensemble scheme will consist of the following sequential steps:

- 1. We perform a non-linear expansion of the inputs for a test sample or one data vector based on (1) and apply it to the first node of the ensemble (*SGD\_1*);
- 2. We add the predicted output from *SGD\_1* as a new independent feature, then perform the procedure (1) and apply it to the second node of the ensemble (*SGD\_2*);
- 3. We add the predicted output from *SGD*\_2 as a new independent feature, then perform the procedure (1) and apply it to the second node of the ensemble (*SGD*\_3);
- 4. .
- 5. We perform similar operations with all the other ensemble nodes until we reach the last one. The prediction result of the last node of the ensemble will be the sought value.

Figure 3 shows a flowchart of the application algorithm of the proposed ensemble scheme using the Wiener polynomial and SGD.



**Figure 3.** Architecture of the proposed ensemble scheme using Wiener polynomial and SGD: (application/test mode).

The following should be noted among the apparent advantages of the proposed scheme:

- Ensuring a high approximation accuracy due to the use of the Wiener polynomial, applied at each step of the ensemble;
- Ensuring the high performance due to the use of SGD as weak predictors;
- The possibility of a high-order approximation of the Wiener polynomial in an implicit form.

The last point is achieved by using a quadratic Wiener polynomial at each node of the ensemble scheme. In addition, each subsequent node of the ensemble uses the result of the work of the previous one. That is, when using the result of the first node of the ensemble (for which quadratic Wiener polynomials are used) in the second node of the ensemble (for which quadratic Wiener polynomials are also used), as a result, we get the fourth order of the polynomial implicitly. Each subsequent node of the ensemble, in the case of using a quadratic Wiener polynomial, doubles the order of the polynomial implicitly compared to the previous one. At the same time, the number of independent attributes grows very slowly compared to the use of a direct approximation by a high-order Wiener polynomial.

This approach provides a high approximation accuracy, similar to the direct approximation by a high-order Wiener polynomial, but without a significant increase in the input data space at each node and works at a high speed.

#### 3. Modeling and Results

The modeling of the new ensemble method took place on an ultrabook with the following parameters: Dell Intel Core i5, RAM 8 GB, and SSD 512 GB. Experimental studies were conducted for a set of biomedical data of a large volume. Let us consider it in more detail.

## 3.1. Dataset Descriptions

This paper solved the problem of predicting the heart rate of a person. We used a real-world dataset from the Kaggle repository [44]. It was formed based on the electrocardiograms of patients with different heart rate levels. The dataset's authors selected several features from the electrocardiograms, the main characteristics of which are presented in Table 1.

Table 1. The main characteristics of the dataset.

Attribute Title	Mean Value	Std	Min Value	Max Value
Mean of RR intervals (MEAN_RR)	845.914	124.485	547.595	1322.01
Median of RR intervals (MEDIANR_R)	841.156	132.003	517.51	1653.12
Standard deviation of RR intervals (SDRR)	109.26	76.8158	27.2406	563.48
Root mean square of successive RR interval differences (RMSSD)	14.9808	4.12688	5.53346	26.6232
Standard deviation of successive RR interval differences (SDRR)	14.9801	4.12688	5.53336	26.623
Ratio of SDRR/RMSSD	7.38995	5.12581	2.66038	54.3399
Percentage of successive RR intervals that differ by more than	0 8/38/	8 20845	0	20 /
25 ms (pNN25)	9.04304	0.20045	0	39.4
Percentage of successive RR intervals that differ by more than	0 86007	0.0021	0	5.4
50 ms (pNN50)	0.00997	0.9921	0	5.4
Kurtosis of distribution of successive RR intervals (KURT)	0.52599	1.78593	-1.8947	62.6724
Skew of distribution of successive RR intervals (SKEW)	0.044	0.69987	-2.1363	6.56471
Mean of relative RR intervals (MEAN_REL_RR)	-0.001	0.00016	-0.0012	0.00123
Median of relative RR intervals (MEDIAN_REL_RR)	-0.0005	0.00087	-0.0044	0.0021
Standard deviation of relative RR intervals (SDRR_REL_RR)	0.01859	0.00547	0.00899	0.03654
Root mean square of successive relative RR interval differences	0.00972	0 00392	0.00322	0.02695
(RMSSD_REL_RR)	0.00772	0.00372	0.00322	0.02075
Standard deviation of successive relative RR interval differences	0.00972	0.00392	0.00322	0.02695
(SDSD_REL_RR)	0.00772	0.00372	0.00322	0.02075
Ratio of SDRR/RMSSD for relative RR interval differences	2 005	0.37551	1 18126	3 70231
(SDRR_RMSSD_REL_RR)	2.000	0.07001	1.10120	00201
Kurtosis of distribution of relative RR intervals (KURT_REL_RR)	0.52599	1.78593	-1.8947	62.6724
Skew of distribution of relative RR intervals (SKEW_REL_RR)	0.044	0.69987	-2.1363	6.56471
Heart rate of the patient at the time of data recorded (HR)	74.0103	10.3811	48.7372	113.727

The dataset contains more than 360,000 observations. The dataset was randomly divided into training (70%) and test samples (30) for the simulation.

# 3.2. Performance Indicators

A number of performance indicators were chosen to evaluate the performance of the proposed ensemble scheme. They will ensure the possibility of performing a comprehensive analysis of the results of the method.

Let us suppose that we have the actual value of the searching attribute and its predicted value  $y_i^{pred}$  by choosing machine learning models for each from the *N* observations in the stated set of data (training or test) i = 1, N. Using this, we can calculate the following performance indicators:

• Maximum residual error (ME):

$$ME\left(y_i^{actual}, y_i^{pred}\right) = \max\left(\left|y_i^{actual} - y_i^{pred}\right|\right),\tag{1}$$

• Median absolute error (MedAE):

$$MedAE\left(y_i^{actual}, y_i^{pred}\right) = \mathrm{median}\left(\left|y_i^{actual} - y_i^{pred}\right|, \dots, \left|y_N^{actual} - y_N^{pred}\right|\right),$$
(2)

• Mean absolute error (MAE):

$$MAE\left(y_{i}^{actual}, y_{i}^{pred}\right) = \frac{1}{N} \sum_{i}^{N} \left|y_{i}^{actual} - y_{i}^{pred}\right|,\tag{3}$$

• Mean square error (MSE):

$$MSE\left(y_{i}^{actual}, y_{i}^{pred}\right) = \frac{1}{N} \sum_{i}^{N} \left(y_{i}^{actual} - y_{i}^{pred}\right)^{2},\tag{4}$$

• Mean absolute percentage error (MAPE):

$$MAE\left(y_i^{actual}, y_i^{pred}\right) = \frac{1}{N} \sum_{i}^{N} \frac{\left|y_i^{actual} - y_i^{pred}\right|}{y_i^{actual}},$$
(5)

• Root mean square error (RMSE):

$$RMSE\left(y_{i}^{actual}, y_{i}^{pred}\right) = \sqrt{\frac{\sum\limits_{i}^{N} \left(y_{i}^{actual} - y_{i}^{pred}\right)^{2}}{N}},$$
(6)

• Coefficient of determination (R2):

$$R2\left(y_{i}^{actual}, y_{i}^{pred}\right) = 1 - \frac{\sum_{i}^{N} \left(y_{i}^{actual} - y_{i}^{pred}\right)^{2}}{\sum_{i}^{N} \left(y_{i}^{actual} - \overline{y}_{i}\right)^{2}}, \quad \overline{y}_{i} = \frac{1}{N} \sum_{i}^{N} y_{i}^{actual}, \tag{7}$$

where  $y_i^{actual}$  is the *i*-th actual value and  $y_i^{pred}$  is the *i*-th predicted value for i = 1, N where N is the number of observations in the dataset.

In addition, since the proposed method is focused on the analysis of large datasets, the ensemble training time *Training\_time* (in seconds) was also taken into account. Actually,

this indicator is the sum of the time of training procedures *time*<sub>l</sub> for the regressors at each of the *l*-th levels of the ensemble:

$$Training\_time = \sum_{k}^{l} time_{k}, \ k = 1, l.$$
(8)

# 3.3. Investigating the Impact of Loss Function on the Prediction Accuracy of the SGD Algorithm

The basis of the proposed ensemble scheme is the use of a regressor based on the SGD algorithm. This choice is justified by its very high performance when analyzing large datasets. As explored in our previous work [33], this machine learning method's accuracy depends on the loss function's choice. The Python library from which we will use the basic implementation of SGD contains four implemented loss functions [33]:

- Epsilon insensitive;
- Huber;
- The squared epsilon insensitive;
- The squared loss.

In order to select the optimal loss function during our dataset analysis, we conducted several experimental studies, the results of which are summarized in Table 2.

Loss Function	ME	MedAE	MAE	MSE	MAPE	RMSE	R2	Training Time, s
			Training	mode				
Huber	21.168	2.291	2.810	13.908	0.037	3.729	0.869	6.61
Epsilon insensitive	15.192	0.749	1.176	3.745	0.016	1.935	0.965	5.06
Squared error	10.454	0.816	1.159	2.811	0.016	1.677	0.974	5.03
Squared epsilon insensitive	9.995	0.808	1.146	2.705	0.016	1.645	0.975	5.05
			Test m	ode				
Huber	20.973	2.300	2.821	14.008	0.037	3.743	0.870	-
Epsilon insensitive	15.150	0.753	1.181	3.745	0.016	1.935	0.965	-
Squared error	10.456	0.821	1.164	2.840	0.016	1.685	0.974	-
Squared epsilon insensitive	9.998	0.814	1.151	2.736	0.016	1.654	0.975	-

Table 2. Performance indicators for different loss functions.

In order to visualize the obtained results, Figure 4 shows the SGD performance errors when using all four loss functions.



Figure 4. Accuracy of the SGD algorithm using different loss functions: (a) R2; (b) RMSE.

As can be seen from Table 2 and Figure 4, the lowest accuracy and, at the same time, the longest training time is demonstrated by the SGD when using the *huber* loss function. The other three loss functions show very close results regarding both the performance accuracy and SGD training time when using them. However, to a small extent, the squared epsilon insensitive loss function stands out among them, demonstrating both the highest accuracy of the work among those considered and a satisfactory time of the training procedure. That is why it was chosen as the primary loss function for the following experiments.

# 3.4. Investigating the Impact of Wiener Polynomial Degree on the Prediction Accuracy and Training Time of the SGD Algorithm

Despite the high training speed, the SGD algorithm is not characterized by a high operation accuracy. In order to eliminate this shortcoming, in [33], it is proposed to perform a non-linear expansion of the inputs with a Wiener polynomial. The authors of [33] experimentally showed that increasing this polynomial order increases the SGD algorithm's accuracy. However, they operated with a short dataset. In this paper, we also conducted some experimental studies on the accuracy of the classical SGD using the quadratic Wiener polynomial. Increasing the order of the Wiener polynomial when processing large volumes of data is not appropriate. In addition to the fact that this will significantly increase the training time of the model, a significant increase in the input data space can cause an overfitting. The results of this experiment are summarized in Table 3.

Method	ME	MedAE	MAE	MSE	MAPE	RMSE	R2	Training Time, s
			Training m	ode				
SGD algorithm	9.995	0.808	1.146	2.705	0.016	1.645	0.975	5.05
SGD algorithm + 2nd degree of Wiener polynomial	5.265	0.276	0.428	0.452	0.006	0.672	0.996	15.81
			Test mod	le				
SGD algorithm	10.038	0.814	1.151	2.741	0.016	1.656	0.975	-
SGD algorithm + 2nd degree of Wiener polynomial	5.250	0.276	0.428	0.454	0.006	0.674	0.996	-

Table 3. Performance indicators for different Wiener polynomial degrees.

In order to visualize the obtained results, Figure 5 shows the dynamics of changes in the SGD operation error and its training time when using the classic SGD in an input expansion scheme with a quadratic Wiener polynomial.

As seen in Table 3, applying the quadratic Wiener polynomial significantly increased the SGD operation's accuracy compared to its accuracy on the original dataset. However, its training time increased from 5 to 15 s.

Experimental studies in [33] were carried out by increasing the power of the Wiener polynomial up to six. However, the dataset used by the authors was small. In our case, we are working with a large dataset. That is why using the Wiener polynomial of higher orders in an explicit form can also increase the prediction accuracy. However, it will significantly complicate the procedure and training time in connection with many attributes in the form of members of the Wiener polynomial of high orders, which will be submitted to the algorithm's input. Since this paper proposes a cascade scheme for the approximation by a Wiener polynomial of high degrees in an implicit form, in further experimental studies, we will stop at the use of a quadratic Wiener polynomial. It provides a sufficient operation accuracy with satisfactory time characteristics of the training procedure, which is a significant point during its further use as part of the proposed ensemble scheme.



**Figure 5.** Influence of the Wiener polynomial degree on the performance of the SGD algorithm: (a) RMSE; (b) training time, seconds.

# 3.5. Investigating the Impact of Cascade Level on the Prediction Accuracy of the Proposed Scheme

Cascade algorithms are characterized by the need to select one crucial parameter: the number of cascade levels [45]. As mentioned in subSection 3.4, the number of nodes in the ensemble scheme developed in this work is set by the user. It can also be implemented automatically until the required accuracy of the method is obtained. In both cases, it is necessary to select the optimal number of levels to receive the highest possible accuracy of the method on the one hand, and the highest generalization properties of the method on the other.

Therefore, we carried out experimental studies to determine the optimal value of this indicator. The results of this experiment are summarized in Table 4.

Level Number of the Proposed Ensemble	ME	MedAE	MAE	MSE	MAPE	RMSE	R2	Training Time, s
			Training	mode				
1	5.265	0.276	0.428	0.452	0.006	0.672	0.996	15.81
2	4.189	0.225	0.313	0.207	0.004	0.455	0.998	4.08
3	8.155	0.228	0.303	0.199	0.004	0.446	0.998	4.09
4	8.019	0.236	0.318	0.211	0.004	0.459	0.998	5.29
			Test m	ode				
1	5.250	0.276	0.428	0.454	0.006	0.674	0.996	-
2	4.266	0.227	0.317	0.213	0.004	0.462	0.998	-
3	10.739	0.228	0.304	0.198	0.004	0.445	0.998	-
4	13.032	0.240	0.323	0.224	0.004	0.474	0.998	-

Table 4. Performance indicators for a different level number of the proposed ensemble.

In order to visualize the obtained results, Figure 6 shows the dynamics of changes in the SGD operation errors when using a different number of levels in the proposed ensemble scheme.



**Figure 6.** Influence of the proposed scheme's levels on the prediction accuracy of the proposed ensemble scheme: (a) MAE; (b) MSE. The *ox*-axis, in this case, indicates the ensemble levels (1, 2, 3, and 4).

As can be seen from Figure 6, the errors of the first level of the developed scheme correspond to the errors of the SDG with a quadratic Wiener polynomial. However, the further increase in the number of levels of the proposed scheme significantly increased the accuracy of its operation. In addition, the training time dropped significantly, according to (9). This is because the training time is calculated as the training procedure duration of both SGDs from the two levels of the ensemble scheme. Each of them, in turn, processes half the amount of data than the SGD of the first level of the proposed scheme. That is why the training time has decreased by more than 3.5 times.

However, in this case, the main advantage is that the two-level developed scheme provided an implicit approximation by the resulting Wiener polynomial of the fourth degree. It happened because the first level of the scheme uses a quadratic polynomial. The result of its work is transmitted and considered by the second-level regressor, which also uses a quadratic Wiener polynomial. As a result, we get a Wiener polynomial of the 4th degree, but without a significant input expansion, as this could happen in the case of a direct use of this polynomial degree.

The use of a three-level scheme further increased the prediction accuracy. In addition, the power of the Wiener polynomial doubled again. The approximation, in this case, took place using the eighth power of the polynomial (again in an implicit form). A further increase in the number of levels of the proposed ensemble scheme shows an increase in all the training and application mode errors. The deterioration of properties explains this before the generalization. That is why the optimal value of the number of nodes of the developed ensemble scheme during the analysis of the dataset we studied is three.

#### 3.6. Results of the Application of the Cascading Scheme Using Ito Decomposition and SGD

Table 5 summarizes the performance indicators of its work in training and test modes based on the selected optimal parameters of the work of the developed ensemble scheme.

<b>Optimal Parameters/Performance Indicators</b>	ME	MedAE	MAE	MSE	MAPE	RMSE	R2	Training Time, s
MinMaxScaler(); quadratic Wiener polynomial;	8.155	0.228	0.303	<i>Traini</i> 0.199	ing mode 0.004	0.446	0.998	4.09
SGD with squared epsilon insensitive loss function; 3 levels of the proposed ensemble scheme	10.739	0.228	0.304	Tes 0.198	t mode 0.004	0.445	0.998	-

Table 5. Performance indicators for the proposed ensemble scheme with optimal parameters.

The results show that the three-level ensemble scheme using the quadratic Wiener polynomial provides a high approximation accuracy and generalization ability when analyzing a sizeable biomedical dataset. In addition, a possible overfitting due to an increase in the number of inputs is not observed.

The developed ensemble scheme with the optimal parameters will be used further to evaluate its effectiveness with the effectiveness of the several existing, most similar methods.

# 4. Comparison and Discussion

4.1. Comparison with Existing Methods

To compare the performance indicators of the proposed ensemble scheme, we chose similar methods from different classes:

- 1. The SGD algorithm [41];
- 2. The SGD algorithm with quadratic Wiener polynomial [33];
- 3. The Gradient Boosting ensemble [46];
- 4. The AdaBoost ensemble [47].

Performance indicators for all the investigated methods in raining and test modes are presented in Table 6.

Method (Test Mode)	ME	MedAE	MAE	MSE	MAPE	RMSE	R2	Training Time, s
Proposed method	10.739	0.228	0.304	0.198	0.004	0.445	0.998	4.089
Gradient Boosting Regressor [46]	6.412	0.227	0.343	0.264	0.005	0.514	0.998	169.547
SGD algorithm + 2nd degree of	5.250	0.276	0.428	0.454	0.006	0.674	0.996	15.810
Wiener polynomial [33]								
SGD algorithm [41]	9.998	0.814	1.151	2.736	0.016	1.654	0.975	5.047
AdaBoost Regressor [47]	5.160	1.518	1.585	3.359	0.022	1.833	0.969	66.531

Table 6. Performance indicators for all investigated methods.

In order to visualize the obtained results, Figure 7 shows the values of the most informative operating errors of all the studied methods in the application mode.

As can be seen from Figure 7, the largest values of all the errors were obtained for regressors based on the AdaBoost and classical SGD. A significantly better result (more than six times smaller MSE) was demonstrated by the SGD using the quadratic Wiener polynomial. Another advantage of using such a combination is improving the accuracy of solving regression problems in the case of large datasets in biomedical engineering. A regressor based on Gradient Boosting demonstrated a slightly better result in terms of the accuracy but a much worse (by more than ten times) training time.

The highest accuracy of solving the stated task was obtained using the developed threelevel ensemble scheme based on the SGD and quadratic Wiener polynomial. In addition, the proposed scheme demonstrates a 41 times faster implementation of the training procedure compared to the nearest competitor in terms of the accuracy. It is even though the Gradient Boosting worked exclusively with the initial dataset, and the developed scheme processed a significantly increased number of features of the set due to the application of the developed Wiener quadratic polynomial scheme at each level.





# 4.2. Limitations of the Proposed Approach

A feature of the non-linear expansion of the inputs by the Wiener polynomial, as a discrete analog of the Voltaire series, is a significant increase in the approximation accuracy. At the same time, the number of features of the dataset represented by members of the Wiener polynomial increases almost unrealistically as its order increases. It imposes several limitations on using this tool in an explicit form during the big data analysis.

The complexity of the models based on the Wiener polynomial can be determined by the number of coefficients near its members. Therefore, the approximation by a high-order polynomial leads to a significant increase in the complexity of the calculations. That is why the main advantage of the developed scheme is the possibility of approximating large datasets with non-linear response surfaces by high orders of the Wiener polynomial in an implicit form. That is, without significantly increasing the number of inputs of the selected regressor. In particular, Figure 8 shows a graph of the change in the number of Wiener polynomial members (red line) generated for a stated set of data when using different powers up to and including eight. Next to it, a green line shows a graph of the change in the number of the attributes of the multilevel developed scheme which, at the third level, also performs an approximation by the Wiener polynomial of the eighth degree (however implicitly).

The graph clearly shows that the number of coefficients that should be searched for when using an approximation by a polynomial of the eighth degree is unrealistically large. At the same time, the developed scheme provides the same result in accuracy with a significant reduction in the computational complexity and training time. It is explained by the fact that the number the polynomial members at each cascade node does not increase significantly. Despite this, the number of inputs using the proposed approach keeps growing compared to the original data set due to their quadratic polynomial expansion. It can be a problem when solving applied biomedical engineering tasks, which are characterized by large volumes of data with a lot of initial input attributes.



**Figure 8.** The number of the Wiener polynomial members during approximation of the stated dataset by different polynomial degrees (2–8) via direct and proposed cascade approximations. The first number in the chart indicates the polynomial degree, and the second number indicates the member's number.

# 4.3. Possibilities for the Future Research

Suppose the response surface of a specific task is significantly non-linear, and the quadratic polynomial, even in the proposed ensemble scheme, does not provide a sufficient prediction accuracy. In that case, using the cubic Wiener polynomial at each cascade step will be appropriate. The advantage of such an approach will be that the cubic polynomial is characterized by higher approximation properties, which will increase the accuracy and cause a significant increase in the independent features of the initial dataset.

That is why, in the perspective of further research, it is necessary to consider the possibility of reducing the number of attributes at each level of the proposed scheme while maintaining the accuracy of its operation.

For this, dimensionality reduction procedures (PCA) can be used based on the neural network tools. Therefore, future research can be done according to the scheme presented in Figure 9.



Figure 9. Future-research ensemble scheme using dimensionality reduction blocks at each level.

In particular, using dimensionality reduction blocks (PCA) at each level of the ensemble scheme will allow one to control the number of independent features before applying them to the selected regressor. It will significantly reduce the training time of the latter.

This approach will improve the effectiveness of using the developed ensemble scheme in the case of many independent attributes of the initial dataset, which a significantly non-linear response surface will characterize.

In addition, among the prospects for further research, it would be good to consider other options for the non-linear expansion of the inputs, as well as the use of different methods as a basic regressor or classifier at each level of the developed ensemble scheme.

#### 5. Conclusions

This paper solved the problem of approximating large datasets in the biomedical engineering area using a machine learning approach. Since single machine learning methods do not provide a sufficient approximation accuracy and high generalization, the authors consider the class of ensemble machine learning. The paper outlines the shortcomings of three main groups of ensemble machine learning methods in the case of the analysis of large datasets. Among them are the high complexity of training procedures, the large computing resources for its implementation, and a considerable duration of its work.

Previous studies [33] demonstrate the high efficiency of approximating non-linear dependencies by the Wiener polynomial. The high-speed implementation of machine learning procedures to implement this approach is based on one of the fastest machine learning methods, SGD. However, increasing the approximation accuracy requires expanding the order of this polynomial, which causes a significant increase in the number of independent features for analysis. This, in turn, leads to an increase in the duration of the training procedures, which is critical when analyzing large datasets. In addition, this approach can provoke an overfitting of the selected machine learning method.

In order to eliminate these shortcomings, the authors developed a new ensemble structure that combines both of the above instruments but avoids the drawbacks of their work. The procedures of its training and application are described and illustrated in detail. The cascade scheme allows for a high-order approximation of the Wiener polynomial without significantly increasing the number of independent features in the dataset.

The modeling was carried out using a real-world dataset of a large volume. The paper presents several results of the experimental studies on selecting the optimal parameters of the developed ensemble scheme. The results of the implicit ensemble approximation by the eighth-order polynomial shows no significant increase in the feature number of the set compared with the direct approximation by the Wiener polynomial of the eighth order. It was established that the developed ensemble scheme demonstrates a 41 times faster learning procedure and almost twice lower errors than the Gradient Boosting method.

Among the disadvantages of all the methods of the cascading class, the need for a sequential execution of all the algorithm steps should be noted, which increases the training procedure time compared with other classes of the ensemble methods. To eliminate this shortcoming, further studies suggest using a PCA at each level of the ensemble scheme. The control of the amount of input data at each node of the ensemble scheme will ensure the preservation of the accuracy of the work while significantly reducing the time of its training.

Author Contributions: Conceptualization, R.T. and I.I.; methodology, I.I.; software, R.H. and K.Y.; validation, M.H. and R.T.; formal analysis, R.T., M.H. and S.K.S.; investigation, I.I. and R.H.; resources, I.I.; data curation, R.H. and K.Y.; writing—original draft preparation, I.I.; writing—review and editing, I.I., K.Y. and M.H.; visualization, R.H.; supervision, R.T.; project administration, S.K.S.; funding acquisition, I.I. All authors have read and agreed to the published version of the manuscript.

**Funding:** The National Research Foundation of Ukraine funded this research under project number 2021.01/0103.

Data Availability Statement: The data supporting this study's findings are openly available in [44].

**Acknowledgments:** The authors thank the reviewers for the correct and concise recommendations that helped present the materials better.

Conflicts of Interest: The authors declare no conflict of interest.

# References

- 1. Garza-Ulloa, J. *Applied Biomedical Engineering Using Artificial Intelligence and Cognitive Models;* Academic Press: London, UK, 2022; ISBN 978-0-12-820934-9.
- Tsmots, I.; Skorokhoda, O. Methods and VLSI-Structures for Neural Element Implementation. In Proceedings of the 2010 VIth International Conference on Perspective Technologies and Methods in MEMS Design, Lviv, Ukraine, 20–23 April 2010; p. 135.
- 3. Teslyuk, V.; Beregovskyi, V.; Denysyuk, P.; Teslyuk, T.; Lozynskyi, A. Development and Implementation of the Technical Accident Prevention Subsystem for the Smart Home System. *Int. J. Intell. Syst. Appl.* **2018**, *10*, 1–8. [CrossRef]
- Radutniy, R.; Nechyporenko, A.; Alekseeva, V.; Titova, G.; Bibik, D.; Gargin, V.V. Automated Measurement of Bone Thickness on SCT Sections and Other Images. In Proceedings of the 2020 IEEE Third International Conference on Data Stream Mining & Processing (DSMP), Lviv, Ukraine, 21–25 August 2020; IEEE: Piscataway, NJ, USA, 2020; pp. 222–226.
- Nechyporenko, A.S.; Radutny, R.; Alekseeva, V.V.; Titova, G.; Gargin, V.V. Complex Automatic Determination of Morphological Parameters for Bone Tissue in Human Paranasal Sinuses. *Open Bioinform. J.* 2021, 14, 130–137. [CrossRef]
- Babichev, S.; Škvor, J. Technique of Gene Expression Profiles Extraction Based on the Complex Use of Clustering and Classification Methods. *Diagnostics* 2020, 10, 584. [CrossRef] [PubMed]
- Mochurad, L.; Yatskiv, M. Simulation of a Human Operator's Response to Stressors under Production Conditions. In Proceedings of the 3rd International Conference on Informatics and Data-Driven Medicine, Växjö, Sweden, 19–21 November 2020; CEUR-WS 2753, pp. 156–169.
- Chumachenko, D.; Chumachenko, T.; Meniailov, I.; Pyrohov, P.; Kuzin, I.; Rodyna, R. On-Line Data Processing, Simulation and Forecasting of the Coronavirus Disease (COVID-19) Propagation in Ukraine Based on Machine Learning Approach. In Proceedings of the Data Stream Mining and Processing, Lviv, Ukraine, 21–25 August 2020; Springer: Cham, Switzerland, 2020; pp. 372–382.
- 9. Krak, I.; Barmak, O.; Manziuk, E. Using Visual Analytics to Develop Human and Machine-centric Models: A Review of Approaches and Proposed Information Technology. *Comput. Intell.* **2020**, *38*, 921–946. [CrossRef]
- Bisikalo, O.; Chernenko, D.; Danylchuk, O.; Kovtun, V.; Romanenko, V. Information Technology for TTF Optimization of an Information System for Critical Use That Operates in Aggressive Cyber-Physical Space. In Proceedings of the 2020 IEEE International Conference on Problems of Infocommunications. Science and Technology (PIC S&T), Kharkiv, Ukraine, 6–9 October 2020; IEEE: Piscataway, NJ, USA, 2020; pp. 323–329.
- Bisikalo, O.V.; Kovtun, V.V.; Kovtun, O.V.; Romanenko, V.B. Research of Safety and Survivability Models of the Information System for Critical Use. In Proceedings of the 2020 IEEE 11th International Conference on Dependable Systems, Services and Technologies (DESSERT), Kyiv, Ukraine, 14–18 May 2020; IEEE: Piscataway, NJ, USA, 2020; pp. 7–12.
- 12. Park, C.; Took, C.C.; Seong, J.-K. Machine Learning in Biomedical Engineering. *Biomed. Eng. Lett.* 2018, *8*, 139–155. [CrossRef] [PubMed]
- 13. Singh, Y.; Tiwari, M. A Novel Hybrid Approach for Detection of Type-2 Diabetes in Women Using Lasso Regression and Artificial Neural Network. *Int. J. Intell. Syst. Appl.* **2022**, *14*, 11–20. [CrossRef]
- 14. Polatgil, M. Investigation of the Effect of Normalization Methods on ANFIS Success: Forestfire and Diabets Datasets. *Int. J. Inf. Technol. Comput. Sci.* 2022, 14, 1–8. [CrossRef]
- 15. Korystin, O.; Nataliia, S.; Mitina, O. Risk Forecasting of Data Confidentiality Breach Using Linear Regression Algorithm. *Int. J. Comput. Netw. Inf. Secur.* 2022, 14, 1–13. [CrossRef]
- 16. Tepla, T. *Biocompatible Materials Selection via New Supervised Learning Methods;* LAP LAMBERT Academic Publishing: Chisinau, Moldova, 2019; ISBN 978-613-9-44384-0.
- 17. Hu, Z.; Ivashchenko, M.; Lyushenko, L.; Klyushnyk, D. Artificial Neural Network Training Criterion Formulation Using Error Continuous Domain. *Int. J. Mod. Educ. Comput. Sci.* **2021**, *13*, 13–22. [CrossRef]
- 18. Hu, Z.; Bodyanskiy, Y.V.; Kulishova, N.Y.; Tyshchenko, O.K. A Multidimensional Extended Neo-Fuzzy Neuron for Facial Expression Recognition. *Int. J. Intell. Syst. Appl.* **2017**, *9*, 29–36. [CrossRef]
- 19. Hu, Z.; Tereykovski, I.A.; Tereykovska, L.O.; Pogorelov, V.V. Determination of Structural Parameters of Multilayer Perceptron Designed to Estimate Parameters of Technical Systems. *Int. J. Intell. Syst. Appl.* **2017**, *9*, 57–62. [CrossRef]
- 20. Babenko, V.; Panchyshyn, A.; Zomchak, L.; Nehrey, M.; Artym-Drohomyretska, Z.; Lahotskyi, T. Classical Machine Learning Methods in Economics Research: Macro and Micro Level Examples. *WSEAS Trans. Bus. Econ.* **2021**, *18*, 209–217. [CrossRef]
- 21. Izonin, I.; Trostianchyn, A.; Duriagina, Z.; Tkachenko, R.; Tepla, T.; Lotoshynska, N. The Combined Use of the Wiener Polynomial and SVM for Material Classification Task in Medical Implants Production. *Int. J. Intell. Syst. Appl.* **2018**, *10*, 40–47. [CrossRef]
- 22. Pandey, H.; Goyal, R.; Virmani, D.; Gupta, C. Ensem\_SLDR: Classification of Cybercrime Using Ensemble Learning Technique. *Int. J. Comput. Netw. Inf. Secur.* 2021, 14, 81–90. [CrossRef]
- 23. Maduranga, M.W.P.; Abeysekera, R. TreeLoc: An Ensemble Learning-Based Approach for Range Based Indoor Localization. *Int. J. Wirel. Microw. Technol.* **2021**, *11*, 18–25. [CrossRef]

- 24. Khan, Z.M. Hybrid Ensemble Learning Technique for Software Defect Prediction. IJMECS 2020, 12, 1–10. [CrossRef]
- Kotsovsky, V.; Geche, F.; Batyuk, A. On the Computational Complexity of Learning Bithreshold Neural Units and Networks. In Proceedings of the Lecture Notes in Computational Intelligence and Decision Making, Salisnyj Port, Ukraine, 21–25 May 2019; Springer: Cham, Switzerland, 2019; pp. 189–202.
- Garza-Ulloa, J. Machine Learning Models Applied to Biomedical Engineering. In Applied Biomedical Engineering Using Artificial Intelligence and Cognitive Models; Elsevier: Amsterdam, The Netherlands, 2022; pp. 175–334, ISBN 978-0-12-820718-5.
- 27. Sajedi, H.; Masoumi, E. Construction of High-Accuracy Ensemble of Classifiers. *Int. J. Inf. Technol. Comput. Sci.* 2014, 6, 1–10. [CrossRef]
- Wu, J.; Chen, S.; Zhou, W.; Wang, N.; Fan, Z. Evaluation of Feature Selection Methods Using Bagging and Boosting Ensemble Techniques on High Throughput Biological Data. In Proceedings of the 2020 10th International Conference on Biomedical Engineering and Technology, Tokyo, Japan, 15 September 2020; ACM: New York, NY, USA, 2020; pp. 170–175.
- 29. Ababor Abafogi, A. Boosting Afaan Oromo Named Entity Recognition with Multiple Methods. *Int. J. Inf. Eng. Electron. Bus.* 2021, 13, 51–59. [CrossRef]
- Mateo, J.; Rius-Peris, J.M.; Maraña-Pérez, A.I.; Valiente-Armero, A.; Torres, A.M. Extreme Gradient Boosting Machine Learning Method for Predicting Medical Treatment in Patients with Acute Bronchiolitis. *Biocybern. Biomed. Eng.* 2021, 41, 792–801. [CrossRef]
- Abuhaiba, I.S.I.; Dawoud, H.M. Combining Different Approaches to Improve Arabic Text Documents Classification. Int. J. Intell. Syst. Appl. 2017, 9, 39–52. [CrossRef]
- Rahman, T.; Chowdhury, M.; Khandakar, A.; Mahbub, Z.B.; Hossain, M.S.A.; Alhatou, A.; Abdalla, E.; Muthiyal, S.; Islam, K.F.; Kashem, S.B.A.; et al. BIO-CXRNET: A Robust Multimodal Stacking Machine Learning Technique for Mortality Risk Prediction of COVID-19 Patients Using Chest X-Ray Images and Clinical Data. *arXiv* 2022, arXiv:2206.07595.
- 33. Izonin, I.; Greguš, M.L.; Tkachenko, R.; Logoyda, M.; Mishchuk, O.; Kynash, Y. SGD-Based Wiener Polynomial Approximation for Missing Data Recovery in Air Pollution Monitoring Dataset. In Proceedings of the Advances in Computational Intelligence, Gran Canaria, Spain, 12–14 June 2019; Rojas, I., Joya, G., Catala, A., Eds.; Springer International Publishing: Cham, Switzerland, 2019; pp. 781–793.
- Group Method of Data Handling (GMDH) for Deep Learning, Data Mining Algorithms Optimization, Fuzzy Models Analysis, Forecasting Neural Networks and Modeling Software Systems. Available online: http://www.gmdh.net/ (accessed on 11 October 2022).
- Lytvynenko, V.; Wojcik, W.; Fefelov, A.; Lurie, I.; Savina, N.; Voronenko, M.; Boskin, O.; Smailova, S. Hybrid Methods of GMDH-Neural Networks Synthesis and Training for Solving Problems of Time Series Forecasting. In *Lecture Notes in Computational Intelligence and Decision Making*; Lytvynenko, V., Babichev, S., Wójcik, W., Vynokurova, O., Vyshemyrskaya, S., Radetskaya, S., Eds.; Advances in Intelligent Systems and Computing; Springer International Publishing: Cham, Switzerland, 2020; Volume 1020, pp. 513–531, ISBN 978-3-030-26473-4.
- Open'ko, P.; Kobzev, V.; Larin, V.; Drannyk, P.; Tkachev, V.; Uhrynovych, O. The Problem Solution of the Surface-to-Air Missile Systems Electronic Equipment Durability Prediction When Implementing the Strategy of Condition-Based Maintenance and Repair Using the Group Method of Data Handling. *Sci. Pap. Social Dev. Secur.* 2021, *11*, 90–97. [CrossRef]
- 37. Ivakhnenko, A.G.; Ivakhnenko, G.A.; Savchenko, E.; Wunsch, D. Problems of Further Development of GMDH Algorithms: Part 2. In *Mathematical Theory of Pattern Recognition*; MAIK "Nauka /Interperiodica": Sankt Petersburg, Russia, 2002.
- Salamh, M.; Wang, L. Second-Order Least Squares Method for Dynamic Panel Data Models with Application. J. Risk Financ. Manag. 2021, 14, 410. [CrossRef]
- 39. Lake, R.W.; Shaeri, S.; Senevirathna, S. Limitations of Parametric Group Method of Data Handling and Empirical Improvements for the Application of Rainfall Modelling; Research Square: Durham, NC, USA, 2022.
- 40. Gatto, M.; Marcuzzi, F. Unbiased Least-Squares Modelling. Mathematics 2020, 8, 982. [CrossRef]
- Ighalo, J.O.; Adeniyi, A.G.; Marques, G. Application of Linear Regression Algorithm and Stochastic Gradient Descent in a Machine-Learning Environment for Predicting Biomass Higher Heating Value. *Biofuels Bioprod. Biorefin.* 2020, 14, 1286–1295. [CrossRef]
- 42. Piltan, F.; Bayat, R.; Mehara, S.; Meigolinedjad, J. GDO Artificial Intelligence-Based Switching PID Baseline Feedback Linearization Method: Controlled PUMA Workspace. *Int. J. Inf. Eng. Electron. Bus.* **2012**, *4*, 17–26. [CrossRef]
- Hu, Z.; Odarchenko, R.; Gnatyuk, S.; Zaliskyi, M.; Chaplits, A.; Bondar, S.; Borovik, V. Statistical Techniques for Detecting Cyberattacks on Computer Networks Based on an Analysis of Abnormal Traffic Behavior. *Int. J. Comput. Netw. Inf. Secur.* 2021, 12, 19–27. [CrossRef]
- Heart Rate Prediction to Monitor Stress Level. Available online: https://www.kaggle.com/vinayakshanawad/heart-rateprediction-to-monitor-stress-level (accessed on 19 June 2022).
- 45. Izonin, I.; Tkachenko, R. An Approach towards the Response Surface Linearization via ANN-Based Cascade Scheme for Regression Modeling in Healthcare. *Procedia Comput. Sci.* 2022, 198, 724–729. [CrossRef]
- 46. Theerthagiri, P. Predictive Analysis of Cardiovascular Disease Using Gradient Boosting Based Learning and Recursive Feature Elimination Technique. *Intell. Syst. Appl.* **2022**, *16*, 200121. [CrossRef]
- Kundu, M.; Nashiry, M.A.; Dipongkor, A.K.; Sarmin Sumi, S.; Hossain, M.A. An Optimized Machine Learning Approach for Predicting Parkinson's Disease. *Int. J. Mod. Educ. Comput. Sci.* 2021, 13, 68–74. [CrossRef]