ROMANIAN JOURNAL OF INFORMATION SCIENCE AND TECHNOLOGY
Volume 27, Number 3–4, 2024, 295–309

Deep Learning-Based Prediction Models for the Detection of Vitamin D Deficiency and 25-Hydroxyvitamin D Levels Using Complete Blood Count Tests

Uğur Engin EŞSIZ[1], Çiğdem İnan ACI[2,*], Esra SARAÇ[3], and Mehmet ACI[2]

[1]Department of Industrial Engineering, Çukurova University, Adana, Turkey
[2]Department of Computer Engineering, Mersin University, Mersin, Turkey
[3]Department of Computer Engineering, Adana Alparslan Türkeş Science and Technology University, Adana, Turkey
Email: enginessiz@gmail.com,
caci@mersin.edu.tr*,esarac@atu.edu.tr,maci@mersin.edu.tr
* Corresponding author

Abstract. Vitamin D (VitD) is an essential nutrient that is critical for the well-being of both adults and children, and its deficiency is recognized as a precursor to several diseases. In previous studies, researchers have approached the problem of detecting vitamin D deficiency (VDD) as a single "sufficient/deficient" classification problem using machine learning or statistics-based methods. The main objective of this paper is to predict a patient's VitD status (i.e., sufficiency, insufficiency, or deficiency), severity of VDD (i.e., mild, moderate, or severe), and 25-hydroxyvitamin D (25(OH)D) level in a separate deep learning (DL)-based models. An original dataset consisting of complete blood count (CBC) tests from 907 patients, including 25(OH)D concentrations, collected from a public health laboratory was used for this purpose. CNN, RNN, LSTM, GRU and Auto-encoder algorithms were used to develop DL-based models. The top 25 features in the CBC tests were carefully selected by implementing the Extra Trees Classifier and Multi-task LASSO feature selection algorithms. The performance of the models was evaluated using metrics such as accuracy, F1-score, mean absolute error, root mean square error and R-squared. Remarkably, all three models showed satisfactory results when compared to the existing literature; however, the CNN-based prediction models proved to be the most successful.

Key-words: 25(OH)D level; classification; deep learning; feature selection; prediction; vitamin D deficiency.

1. Introduction

Vitamin D (VitD) is one of the essential vitamins for both adult and child health, and its deficiency is seen as a risk factor for many diseases [1]. Many medical conditions are thought to have a basis in VitD Deficiency (VDD) [2]. Because of this, a proper estimate of this deficiency is crucial for patients as well as clinicians. This prediction makes it possible to get rapid results without laboratory testing, allowing the clinician to effectively guide the patient and enabling the patient to take the necessary precautions in a timely manner.

Based on external and internal data, VDD prediction aims to determine a patient's VitD status or 25-hydroxyvitamin D (25(OH)D) level. Since multivariate inputs are preferred for highly accurate results, models that are more flexible are needed to make predictions. In recent years, the development of high-precision prediction models has been a focus of research in various fields:

Pozna and Precup [3] explored a novel framework for modelling observational processes in the context of cognitive processes rooted in a new pattern of human knowledge, analyzing its cultural origins from philosophical, psychological and linguistic perspectives. Borlea et al. [4] proposed a method for improving clusters generated by the K-means algorithm by employing a supervised learning algorithm in a post-processing step, with an emphasis on improving cluster quality rather than reducing processing time. Ren et al. [5] presented a medical image segmentation network with an attention mechanism and a multi-feature fusion structure, which incorporates a convolutional group encoder module and a self-attention module to enhance multi-scale information acquisition, improve edge details, and enable accurate segmentation of medical images. Zamfirache et al. [6] conducted a performance analysis of three control system structures that combine Reinforcement Learning and Metaheuristic Algorithms to optimize control problems, using the Gravitational Search Algorithm for parameter initialization in Deep Q-Learning. The prediction of many parameters, such as the diagnosis of Parkinson's disease [7], Alzheimer's disease [8], tumor segmentation [9], early severity detection of COVID-19 cases [10] and government statistics on COVID-19 [11] have been studied.

When the previous studies on VDD prediction are analyzed, it can be seen that models based on statistics [12], [13] and Machine Learning (ML-based) have been carried out in recent years: Sluyter et al. [1] developed ML-based prediction models and compared their performance with the statistical based models. Carretero et al. [14] used a dataset obtained from 1002 hypertension patients at a Spanish university hospital to determine the VDD status. Support Vector Machine (SVM), RF, NB, and Extreme Gradient Boost (EGB) algorithms were used to develop classification models. Gonoodi et al. [15] proposed a Decision Tree (DT) based prediction model to assess the risk factors for VDD. According to the results, blood Zn levels are a significant risk factor for identifying individuals with VDD among Iranian adolescent girls. Karamizadeh et al. [16], Amiri et al. [17], Padmaja et al. [18], Guo et al. [19], and Bechrouri et al. [20] used ML-based and data mining algorithms to build models for predicting VDD.

Eşsiz et al. [21] predicted VDD in diabetic patients using data mining techniques and feature selection based on historical electronic health records, achieving a classification accuracy of 97.044% using the SVM model with radial kernel and 18 selected features. The motivation for this paper stems from the main shortcomings of Eşsiz et al.'s study reported in [21], and these can be outlined as follows:

- The detection of VDD without the need for costly laboratory tests is an important health indicator, not only for people with diabetes, but also for the general population.

- Categorizing VDD severity into only two classes, ignoring both VitD status and 25(OH)D level, led to a simplistic assessment under a single model; these scenarios need to be investigated independently and thoroughly.

- While ML and data mining are established techniques, DL, as a subset of ML, enhances power and flexibility by representing the world as a nested hierarchy of concepts. Utilizing DL-based methods in VDD prediction models increases their effectiveness and adaptability.

In this paper, an original dataset consisting of Complete Blood Count (CBC) tests of 907 patients (including 25(OH)D concentrations) from a public health laboratory in Turkey was collected to develop three groups of VDD prediction models using DL-based methods:

Model 1: The first model predicts the VitD status of the patient (i.e., sufficiency, insufficiency, or deficiency). **Model 2**: The second model predicts the severity of the patient's VDD (i.e., mild, moderate, or severe). **Model 3**: The third model predicts the patient's 25(OH)D level (e.g., it has been handled as a regression problem). DL-based prediction models combined with feature selection algorithms (i.e. Extra Trees Classifier (ETC) and multi-task LASSO) have been proposed to accurately estimate 25(OH)D levels from CBC test results while reducing the necessary parameters for the prediction models. In line with this goal, individual and hybrid comparisons of the methods were performed using different classification, regression, and feature selection methods.

The contributions of this paper are listed as follows:

- Most VDD detection studies have addressed the problem as a single "sufficient/deficient" classification. This approach gives superficial results in assessing VitD status and VDD severity. The present paper addresses both problems as separate multi-class models.

- To the best of authors' knowledge, this is the first paper to predict VitD status, VDD severity, and 25(OH)D levels as separate models using CBC tests.

- Most of the literature on VDD prediction is based on statistical and ML-based methods, and there is a lack of studies investigating the performance of DL-based models. All three models we developed gave satisfactory results using well-known performance metrics.

- In addition to the development of DL-based models, studies were also carried out to find the parameter that has the most influence on the result in CBC tests, and the top 25 features were selected for each model.

- CBC tests are widely used in the diagnosis of many medical conditions. The remarkable ability of these tests to provide accurate predictions of VitD status and levels through advanced DL-based models promises to reduce both laboratory costs and time spent on testing procedures.

The remainder of this paper is organized as follows. The second section provides a detailed exposition of the research methodology used in this paper, which includes both data analysis and DL-based prediction algorithms. The third section presents the research results and a performance comparison of the algorithms. Concluding remarks are summarized in the fourth section.

2. Material and Methods

This section introduces some issues related to the methodology used in this paper, including data preprocessing, feature selection methods, and DL-based prediction models.

2.1. Data source

The retrospective data used in this paper consisted of CBC tests including 25(OH)D concentrations collected from a public health laboratory in southern Turkey between February 2016 and April 2019. CBC (a.k.a. full blood count) test provides information about the blood and general health. Table 1 shows the basic statistics of the dataset describing the minimum, maximum, and standard deviation of each attribute with normal ranges. The distribution of CBC tests with respect to VitD status and VDD, as well as the age distribution of the patients, are presented in the supplementary file [22].

2.2. Data processing

The data preprocessing was carried out through a three-step procedure: The first step involved data imputation, which focused on addressing missing values before analyzing the data. To address this, Multivariate Imputation by Chained Equations (MICE) algorithm [23] (a.k.a. iterative data imputation) was employed. Due to the imbalanced nature of the dataset, the second step of the data preprocessing involved addressing data balancing. For this purpose, the Synthetic Minority Oversampling Technique [24] (SMOTE) was employed for both Model 1 and Model 2. The third step, feature selection, is a fundamental preprocessing technique commonly employed in most prediction models. For this task, Randomized Decision Trees (a.k.a. the ETC method) was utilized, which belongs to the ensemble classifier family and incorporates the RF algorithm in Model 1 and Model 2. By using training examples, the ETC method calculates the importance of features that affect the model's output [25]. Model 3 utilized the multi-task LASSO technique [26], a widely adopted method for feature selection in datasets with numerous dimensions, as the VitD Dataset contained 51 features (except 25(OH)D).

Table 1. Basic statistics of the dataset with normal ranges (SD=Standard Deviation)

Attribute Index	Attributes	Normal Range (Measurement Unit)	Mean ±SD
0	Gender	Male: 398 (43.89%) Female: 509 (56.11%)	
1	Age	0-100	43.85 ± 19.54
2	Glycated Hemoglobin (HbA1c)	4-6.2 (%)	6.26 ± 4.01
3	HbA1c International Federation of Clinical Chemistry Unit (HbA1c (IFCC))	20-42 (mmol/mol)	42.83 ± 13.30
4	Creatinin	0.84-1.25 (mg/dL)	0.77 ± 0.21
5	Urea	17-43 (mg/dL)	28.89 ± 9.57
6	Uric Acid	3.5-7.2 (mg/dL)	5.35 ± 1.44
7	Alanine Transaminase (ALT)	7-40 (U/L)	21.80 ± 14.05
8	Aspartate Aminotransferase (AST)	9-45 (U/L)	21.24 ± 9.15
9	Alkaline Phosphatase (ALP)	30-120 (U/L)	70.59 ± 26.68
10	Gamma-Glutamyl Transferase (GGT)	0-55 (U/L)	22.55 ± 36.32
11	Sodium	136-146 (mEq/L)	138.90 ± 2.63

Attribute Index	Attributes	Normal Range (Measurement Unit)	Mean ±SD
12	Potassium	3.5-5.1 (mEq/L)	4.51 ± 0.40
13	Total Cholesterol	120-200 (mg/dL)	195.83 ± 45.52
14	High Density Lipoprotein (HDL)	40-60 (mg/dL)	47.72 ± 12.13
15	Low Density Lipoprotein (LDL)	0-130 (mg/dL)	123.18 ± 34.27
16	Triglyceride	40-160 (mg/dL)	171.03 ± 122.92
17	Iron	40-160 (µg/dL)	78.04 ± 37.68
18	Iron Binding	155-355 (µg/dL)	260.00 ± 75.34
19	Calcium	8.8-10.6 (mg/dL)	41.38 ± 544.22
20	Magnesium	1.3-2.7 (mg/dL)	2.07 ± 0.33
21	Creatine Kinase (CK)	30-294 (U/L)	100.50 ± 114.36
22	C-reactive protein (CRP)	8-10 (mg/L)	0.93 ± 2.03
23	White Blood Cell (WBC)	4-11 (x10ˆ3/µL)	7.03 ± 1.88
24	Red Blood Cell (RBC)	3.8-5.8 (x10ˆ6/µL)	4.67 ± 0.47
25	Hemoglobin (HGB)	11.5-16.5 (g/dL)	13.10 ± 1.48
26	Hematocrit (HCT)	37-47 (%)	40.08 ± 4.06
27	Mean Corpuscular Volume (MCV)	78-100 (fL)	86.12 ± 6.92
28	Mean Corpuscular Hemoglobin (MCH)	27-32 (pg)	28.30 ± 3.86
29	Mean Corpuscular Hemoglobin Concentration (MCHC)	30-35 (g/dL)	33.24 ± 13.33
30	Red Blood Cell Distribution Width (RDW-CV)	11.5-14.5 (%)	14.10 ± 1.51
31	Platelet (PLT)	150-400 (x10ˆ3/µL)	265.02 ± 66.59
32	Mean Platelet Volume (MPV)	8-12 (fL)	9.50 ± 1.21
33	Procalcitonin (PCT)	0.12-0.36 (%)	0.25 ± 0.06
34	Platelet Distribution Width (PDW)	25-65 (%)	50.04 ± 7.78
35	Lymphocytes (LY)	1.5-4 (x10ˆ3/µL)	2.25 ± 1.97
36	Monocytes (MO)	0-0.8 (x10ˆ3/µL)	0.41 ± 0.15
37	Neutrophils (NE)	2-7.5 (x10ˆ3/µL)	4.10 ± 1.52
38	Eosinophil (EO)	0-0.4 (x10ˆ3/µL)	0.20 ± 0.15
39	Basophils (BA)	0-0.1 (x10ˆ3/µL)	0.04 ± 0.13
40	MO%	0-10 (%)	5.81 ± 1.27
41	NE%	40-75 (%)	57.39 ± 8.83
42	EO%	0-6 (%)	2.89 ± 1.91
43	BA%	0-2 (%)	0.54 ± 0.29
44	LY%	20-50 (%)	32.19 ± 16.32
45	Sedimentation	0-20 (mm)	11.32 ± 9.41
46	Triiodothyronine (T3)	0.92 -2.76 (nmol/L)	3.17 ± 0.54
47	Thyroxine (T4)	0.7-1.48 (ng/dL)	1.31 ± 4.00
48	Thyroid Stimulating Hormone (TSH)	0.35-4.94 (uIU/mL)	2.41 ± 3.22
49	Ferritin	10-291 (ng/mL)	49.23 ± 53.61
50	Vitamin B12 (VB12)	187-883 (pg/mL)	372.35 ± 233.22
51	25(OH)D	20-40 (ng/mL)	23.72 ± 16.53

2.3. DL-based prediction models

A DL-based prediction model is a type of artificial intelligence model that uses Deep Neural Networks (DNN) to learn and make predictions from data. These models have gained significant popularity and achieved state-of-the-art results in various domains due to their ability to automatically learn representations from raw data and capture complex patterns and relationships [27].

There are two training phases in all DL-based prediction models: feed-forward and back-propagation (BP). DNNs use weight modification as a learning strategy. BP is a strategy for

updating weights and training DNNs [28]. In BP, errors are sent to hidden layers to train the network. Each neuron has an activation function that can help with the training problem. In general, the mathematical method of DNN can be described as follows:

First step, weight initialization. Weight initialization is a procedure to set the weights of a neural network to small random values that define the starting point for the optimization (learning or training) of the neural network model.

Second step, error calculation as (e), which can be done by Mean Squared Error (MSE). MSE compares the predicted output with the true output, which can be a single number or a vector of numbers. The MSE penalizes the DNN for deviating from the true output by squaring the difference between them. It is calculated as the average of the squared differences between the predicted output (P) and the true output (O)

$$e = \frac{1}{n}\left[\sum_{i=1}^{n}(O_i - P_i)^2\right] \tag{1}$$

Third step, reducing the error by adjusting the weights as follows:

$$w(n) = -\eta\frac{\partial e}{\partial w} + \alpha w(n-1) \tag{2}$$

$$\Delta w = -\eta\frac{\partial e}{\partial w} \tag{3}$$

where, w stands for weight, η and α are constant. The weight is adjusted according to the difference between the weighed value and the network output, specified as the error. All the above procedures are common DNN steps. Depending on the model selected, the other steps will be affected. The following sections provide brief information on the DNN architectures used in this paper.

2.3.1. Convolutional Neural Network (CNN)

A Convolutional Neural Network (CNN) is a Feed Forward Neural Networks (FFNN) consisting of convolutional operations and a depth structure and is a typical algorithm for DL [29]. Deep CNNs perform as well in traditional classification and regression problems as they do in image classification tasks [30]. An illustration of the components of a CNN, encompassing an input layer, a convolutional layer, an activation function, a pooling layer, and a fully connected layer given in [22]. The convolutional layer, often referred to as the convolutional kernel, serves as the core of the neural network. It is usually the first layer in a CNN. A convolutional layer has the main advantage of weight distribution [31]. This step can be represented mathematically in terms of:

$$X_m^L = f(\sum_m x_m^{L-1} * W_{mn}^L) + b_n^L \tag{4}$$

where m and n are the dimensions of the convolution filter, L represented the layer, x represented the features, W represents the weight, and b represents the basis, * signifies the convolutional operation and f is the activation function, which may be a sigmoid, a rectified linear unit, or a hyperbolic tangent (tanh). Activation methods can be modified based on the type of data [31].

In addition, the features extracted by the convolution layer are connected to the next layer (pooling layer). The pooling layers are generally intermediate layers between the convolution

and fully connected layers. The pooling layer reduces the number of training parameters and computational cost, and controls overfitting [31]. The output of the pooling layer can be represented mathematically as follows:

$$x_n^L = down\left(x_m^{L-1}\right) \tag{5}$$

where $down\ (\cdot)$ is a type of pooling operation and can be max pooling or average pooling, etc. Once this process is complete, the extracted features from multiple convolutions and pooling layers are connected to the fully connected layers. These are simply FFNNs. They are also the last few layers in the network structure of a CNN. The output of a fully connected layer can be represented using:

$$(y = 1|x; w) = \frac{1}{1 + exp(-w^T x)} \tag{6}$$

where y represents the labels of the data, $x \in R^{K+1} \times 1$ denotes the feature vector in K dimension, $w \in R^{K+1} \times 1$ denotes the parameters of the weight vector.

2.3.2. Recurrent Neural Network (RNN)

In contrast to conventional FFNNs, RNNs incorporate recurrent connections that establish self-connections to capture temporal dynamics. These connections exhibit two key distinctions compared to traditional networks [32]: (1) the nodes within the same hidden layer of an RNN are interconnected, and (2) the inputs to the hidden layer at the current time step consist of both the outputs from the input layer at the current time step and the hidden layer at the previous time step. The fundamental structure of an RNN is depicted in [22]. The symbol x_i represents the inputs at time step t, S_t represents the outputs of the hidden layer at time step t (referred to as the memory at time t), y_t represents the outputs of the output layer at time step t, and [W,U,V] represents shared parameters (where W denotes the weights of inputs, U denotes the weights of inputs at the current state, and V denotes the weights of outputs) [33]. The modeling process is described by:

$$S_t = f(Ux_t + WS_{t-1} + b_h), y_t = f(VS_t + b_0) \tag{7}$$

Here, f(.) denotes an activation function, and b_h and b_0 represent the biasing vectors of the hidden layer and the output layer, respectively.

2.3.3. Long Short-Term Memory (LSTM)

LSTM is a specialized RNN-based algorithm that assumes the role of conventional neurons in hidden layers, referred to as memory blocks. It performs information updates and checks through the utilization of forgetting gates within these memory blocks [34].

Differing from FFNNs, LSTM incorporates feedback connections, enabling it to process entire sequences of data rather than just individual data points. When compared to other variants of RNN, the LSTM, which solely computes memory units using distinct activation functions, boasts a straightforward architecture. The structure of the LSTM unit is visually depicted in [22].

The LSTM unit consists of three key components referred to as gates, responsible for regulating the flow of information in and out of the memory cell or LSTM cell. The first gate is termed the Forget gate, the second is known as the Input gate, and the final one is the Output gate. Combining these three gates with a memory cell or LSTM cell forms an LSTM unit that

can be likened to a layer of neurons in a traditional FFNN, with each neuron comprising a hidden layer and a current state [28].

2.3.4. Gated Recurrent Units (GRU)

A GRU has the adaptive ability to capture the dependencies of different time scales within each recurrent unit. Compared to the LSTM, the GRU introduces two significant improvements: (1) it synthesizes the forget gate and input gate of the LSTM into a single update gate, and (2) it combines the cell state and hidden state [33]. The architectural layout is shown in [22]. The modelling process of the GRU can be briefly illustrated as follows [35]:

Step 1: Disregard inconspicuous information from the previous hidden states and the current inputs, which is determined by the reset gate according to

$$r_t = \sigma(W_r \cdot [h_{t-1}, x_t] + b_r) \tag{8}$$

Here, "." stands for dot-product operation, b_r denotes the bias vector, and its function is akin to that in LSTM, where a smaller value of r_t allows for less information to be propagated. The new candidate memory is then computed using:

$$z_t = \sigma(W_r \cdot [h_{t-1}, x_t] + b_z) \tag{9}$$

Here, b_z represents the bias vector.

Step 2: Control the removal and addition of old and new information, which is governed by the update gate following:

$$\tilde{h_t} = tanh(W \cdot [r_t * h_{t-1}, x_t] + b_h) \tag{10}$$

where "*" stands for multiplication, b_h is the bias vector. Finally, the new memory state is obtained using

$$h_t = (1 - z_t) * h_{t-1} + z_t * \tilde{h_t} \tag{11}$$

In general, both LSTMs and GRUs effectively preserve essential features through various gate functions, guaranteeing the preservation of information during long-term propagation. Consequently, their gate control mechanisms address the vanishing gradient problem encountered in traditional RNNs [34], [35].

2.3.5. Autoencoders (AE)

An AE functions as a three-layer FFNN, comprising an input layer, hidden layer, and output layer. Notably, the input layer and output layer consist of an equal number of neurons, allowing the AE to learn the reconstruction of the provided input. The primary purpose of the hidden layer is to encode the input vector from the input layer into meaningful code, representing distinctive features. Training the AE involves adjusting the number of neurons in the hidden layer in relation to those in the input layer. Throughout the training phase, the input vector is transformed into these features. By introducing labeled data during the training process, AE can be adapted to handle classification tasks. Similarly, for regression tasks, the AE can be modified by altering the output layer to produce continuous values [36].

See [22] for an illustration of an AE as a three-layer network, with x and z represented as follows:

$$Z = h^{(1)}(W^{(1)}x + b^{(1)}) \tag{12}$$

The output layer, denoted by:

$$X' = h^{(2)}(W^{(2)}z + b^{(2)}) \tag{13}$$

where, the same dimensions as the input layer (x') used for input reconstruction. To evaluate the error between the original input x and the reconstructed input x', the AE employs

$$Loss_{(W,b,W',b')} = \sum_{i=1}^{n} ||x_i - x_i'||^2 \tag{14}$$

which corresponds to either the squared Euclidean distance or the cross-entropy loss as follows [28]:

$$Loss_{(x,x')} = -\sum_{i=1}^{x} x_i log(x_i') + (1 - x_i) log(1 - x_i') \tag{15}$$

All of the techniques described in the previous sections were developed using the Python Scikit Learn library [37] and TensorFlow [38] within the Google Colaboratory [39] environment.

3. Results and Discussions

The block diagram and pseudo codes of the DL-based prediction models and parameter settings of the methods are given in the supplementary document of the paper [22]. Before testing the models, missing data were imputed using the MICE algorithm, and the class imbalance problem was solved using the SMOTE algorithm, as described in Section 2.2. The dataset is split into appropriate input-output data for each prediction model. Following this step, the features that most affected the models' output were obtained using the ETC technique for Model 1 and Model 2, and the multi-task LASSO technique for Model 3. Once the features were extracted, the DL models used in this paper were trained and evaluated using a 5-fold cross-validation technique, with 80% of the data used for training and 20% for testing.

Attaining higher accuracy, improved efficiency, and favorable convergence heavily relies on the optimization of parameters. We have used grid search [40] to optimize each hyper-parameter while maintaining the remaining variables as constants. Subsequently, we selected the most suitable parameter values, striking a balance between performance metrics and computational time. See [22] for a table of all parameter settings for the methods.

3.1. Performance metrics

A complexity matrix is used as part of the model performance assessment process and has four distinct variables: True Positive (TP), True Negative (TN), False Positive (FP), and False Negative (FN) [41]. The Accuracy of a prediction model is determined using:

$$Accuracy(\%) = (TP + TN)/(TP + TN + FP + FN) \tag{16}$$

Precision measures the classifier's ability to correctly predict relevant data as follows:

$$Precision(P) = TP/(TP + FP) \tag{17}$$

while Recall assesses the proportion of relevant data identified by the classifier using:

$$Recall(R) = TP/(TP + FN) \tag{18}$$

The F1-score is a metric employed to evaluate the model's performance, calculated by taking the harmonic mean of Precision and Recall denoted by:

$$F1 - Score = 2 \cdot Precision \cdot Recall/(Precision + Recall) \tag{19}$$

Mean Absolute Error (MAE) is a common way to measure how accurate a regression model is. It essentially defines how far off the model's predictions are from the actual values, on average:

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|y_i - y_i'| \tag{20}$$

Root Mean Square Error (RMSE) serves as an alternative metric for evaluating a regression model's goodness of fit. Analogous to MAE, RMSE quantifies the average magnitude of the prediction errors. However, unlike MAE which utilizes absolute differences, RMSE incorporates a squared term. This squared term amplifies the influence of larger residuals, placing greater emphasis on substantial deviations between predicted and observed values.

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - y_i')^2} \tag{21}$$

R-squared (R^2) is a statistical value in regression analysis that tells you how well a model explains the variability in the data. It ranges from 0 to 1, with a higher R^2 indicating a better fit between the model's predictions and the actual observations.

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_i - y_i')^2}{\sum_{i=1}^{n}(y_i - \bar{y_i})^2} \tag{22}$$

where y_i is the predicted value, y_i' is the observed value, and $\bar{y}$ is the mean value of y.

3.2. Feature selection

This paper employs feature importance to identify the factors influencing the target variable. This approach not only enhances the model's performance but also significantly improves its execution time by excluding unnecessary features. In this paper, the top 25 features were selected to ensure that the models worked optimally, striking the right balance between feature inclusion and performance output. The prominent features selected by ETC in Model 1 and Model 2. Since Model 3 is a regression model, the multi-task LASSO approach was used to identify the 25 most optimal features for estimating the patient's 25(OH)D level. The best attribute indexes and feature importance distribution of the models is given in [22].

3.3. Performance results of the prediction models

Fig. 1.a presents the Accuracy and F1-Score outcomes of Model 1 employing various methods, namely CNN, RNN, LSTM, GRU, and AE, both with and without ETC. Throughout all

the obtained results, predictions employing feature selection consistently showed superior performance compared to those without it. Notably, the CNN model with ETC showed the highest Accuracy and F1-Score, achieving values of 0.9630 and 0.9757, respectively, while the AE algorithm displayed comparatively lower performance than the others. Among the considered methods, the RNN approach in conjunction with ETC achieved the second-best performance with an Accuracy of 0.9490 and an F1-Score of 0.9416. Subsequently, the LSTM method with ETC followed closely as the third-best, showcasing an Accuracy of 0.9384 and an F1-Score of 0.9509, in the prediction of the patient's VitD status. Additionally, the performance outcomes of the DL-Based prediction models exhibited close proximity, and further enhancements were observed through the implementation of the ETC algorithm.

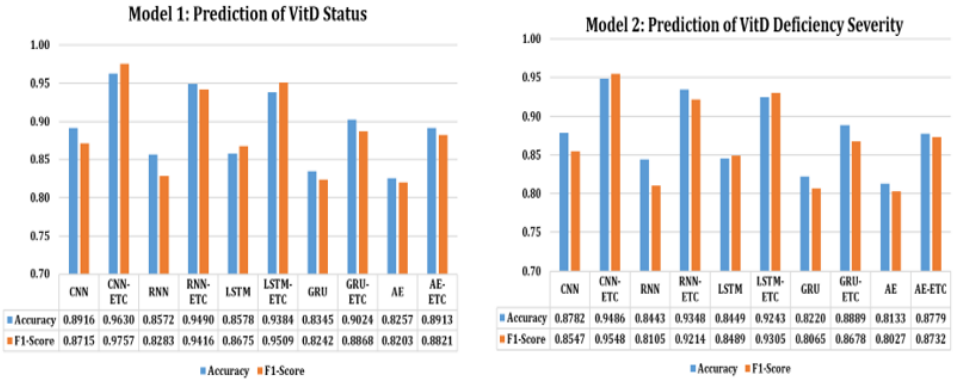

Fig. 1. Accuracy and F1-Score results of Model 1 (a), Model 2 (b).

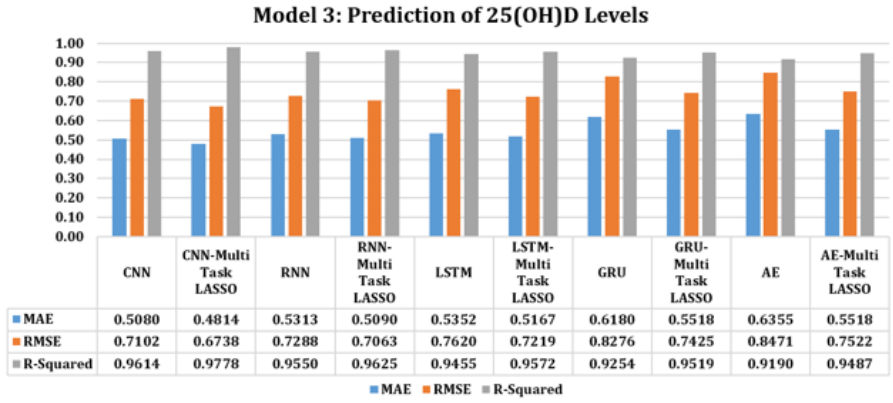

Fig. 2. MAE, RMSE and R-Squared results of Model 3.

Fig. 1.b shows the Accuracy and F1-Score outcomes of Model 2. Similar to Model 1, the CNN-ETC method proves to be the best-performing model, achieving an Accuracy of 0.9486 and an F1-Score of 0.9548. Correspondingly, the relative performance rankings of the other methods remain unchanged. RNN-ETC secures the second position with an Accuracy of 0.9348

and an F1-Score of 0.9214, while LSTM-ETC follows as the third-best method, demonstrating an Accuracy of 0.9243 and an F1-Score of 0.9305. The performances of the GRU and AE models, along with ETC, align closely with the aforementioned three. The close resemblance in results between Model 1 and Model 2 can be attributed to the fact that the data utilized in Model 2 constitutes a subset of that employed in Model 1.

Model 3, dedicated to the prediction of 25(OH)D levels, presents a distinct regression model compared to the preceding two models, as highlighted in Fig. 2. Given its regression nature, the model evaluation employed MAE, RMSE, and R-Squared metrics. For feature selection, a well-established Multi-task LASSO method was employed, yielding performance improvements across all DL-based methods. Notably, despite the differences in performance criteria and feature selection methods, the performance ranking of DL-based methods remained unaltered in Model 3. The CNN-Multi-task LASSO method emerged as the most favorable approach, attaining the lowest MAE (0.4814) and RMSE (0.6738), as well as the highest R-squared value (0.9778). Following closely were the Multi-task LASSO-enhanced versions of RNN, LSTM, GRU, and AE, maintaining their relative performance positions.

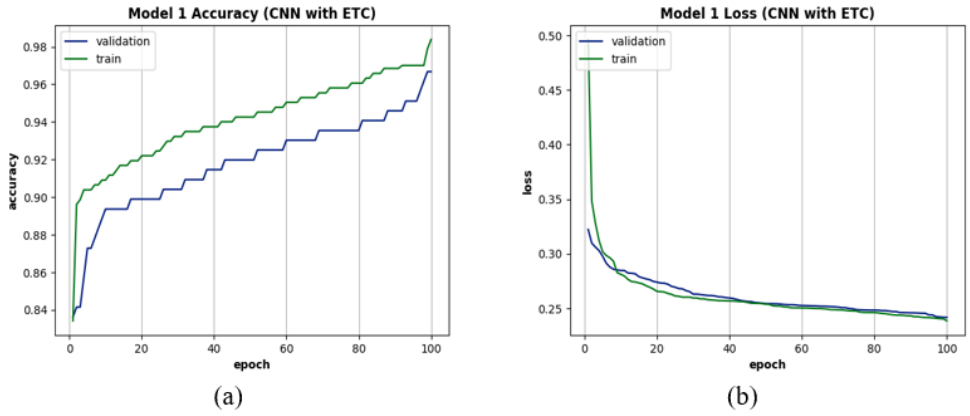

Fig. 3. Convergence plots of Model 1: (a) Accuracy, (b) Loss.

Fig. 3 shows convergence plots for training and validation of Model 1. The training learning curve calculated from the training dataset gives an idea of how well the model is learning. The validation learning curve, calculated from average of the 5-fold cross-validation dataset, gives an idea of how well the model generalizes. All models have been successfully trained and tested. As Model 1 is the model with the highest performance results, the convergence graphs are shown in Fig. 3, and the convergence graphs of Model 2 and Model 3 are shown in the supplementary document [22].

4. Conclusions

The primary goal of this paper was to develop DL-based models for predicting VitD status, VDD severity, and 25(OH)D levels using CBC tests. To achieve this objective, the paper analyzed three sets of prediction models: (1) Model 1, which forecasts the patient's VitD status as sufficient, insufficient, or deficient; (2) Model 2, which predicts the severity of VitD deficiency,

categorized as mild, moderate, or severe; and (3) Model 3, which estimates the patient's 25(OH)D level, treated as a regression problem. To conduct the research, a dataset containing 907 CBC test results was utilized, with 4.4% pertaining to children (aged $\leq$ 10), 8.5% to adolescents (aged 11-17), 86% to adults (aged 18-64), and 1.1% to elders (aged 65+). Prior to model development, the dataset underwent three stages of preprocessing, involving the MICE algorithm, data balancing through the SMOTE technique, and feature selection using ETC and multi-task LASSO techniques. The results demonstrated that all three models surpassed the performance of ML-based or statistics-based approaches previously suggested in the literature.

Acknowledgements. The authors would like to thank the Republic of Turkey Ministry of Health for letting us use CBC tests.

References

[1] J. D. SLUYTER, Y. RAITA, K. HASEGAWA, I. R. REID, R. SCRAGG and C. A. CAMARGO, *Prediction of vitamin D deficiency in older adults: the role of machine learning models*, The Journal of Clinical Endocrinology and Metabolism **107**(10), 2022, pp. 2737–2747.

[2] Y. XU, D. J. BAYLINK, C. S. CHEN, M. E. REEVES, J. XIAO, C. LACY, E. LAU and H. CAO, *The importance of vitamin D metabolism as a potential prophylactic, immunoregulatory and neuroprotective treatment for COVID-19*, Journal of Translational Medicine **18**, 2020, paper 322.

[3] C. POZNA and R.-E. PRECUP, *Aspects concerning the observation process modelling in the framework of cognition processes*, Acta Polytechnica Hungarica **9**(1), 2012, pp. 203–223.

[4] I.-D. BORLEA, R.0E. PRECUP and A.-B. BORLEA, *Improvement of k-means cluster quality by post processing resulted clusters*, Procedia Computer Science **199**, 2022, pp. 63–70.

[5] X. REN, H. LIANG and S. ZHAO, *An attention mechanism and multi-feature fusion network for medical image segmentation*, Proceedings of the Romanian Academy Series A **24**(2), 2023, pp. 191–201.

[6] I. A. ZAMFIRACHE, R.-E. PRECUP and E. M. PETRIU, *Q-learning, policy iteration and actor-critic reinforcement learning combined with metaheuristic algorithms in servo system control*, Facta Universitatis, Series: Mechanical Engineering **21**(4), 2023, pp. 615–630.

[7] M. GERGER and A. GÜMÜşÇÜ, *Diagnosis of Parkinson's disease using spiral test based on pattern recognition*, Romanian Journal of Information Science and Technology **25**(1), 2022, pp. 100–113.

[8] Z. HU, Z. WANG, Y. JIN and W. HOU, *VGG-TSwinformer: transformer-based deep learning model for early Alzheimer's disease prediction*, Computer Methods and Programs in Biomedicine **229**, 2023, paper 107291.

[9] Q. ZHANG, Y. LIANG, Y. ZHANG, Z. TAO, R. LI and H. BI, *A comparative study of attention mechanism based deep learning methods for bladder tumor segmentation*, International Journal of Medical Informatics **171**, 2023, paper 104984.

[10] S. ÖĞÜTCÜ, M. İNAL, C. ÇELIKHASI, U. YILDIZ, N. Ö. DOĞAN and M. PEKDEMIR, *Early detection of mortality in COVID-19 patients through laboratory findings with factor analysis and artificial neural networks*, Romanian Journal of Information Science and Technology **25**(3-4), 2022, pp. 290–302.

[11] S. M. ABRAMOV, S. TRAVIN, G. DUCA and R.-E. PRECUP, *New opportunities model for monitoring, analyzing and forecasting the official statistics on coronavirus disease pandemic*, Romanian Journal of Information Science and Technology **26**(1), 2023, pp. 48–63.

[12] C. S. KHOO, M. F. SHUKOR, J. K. TAN, M. M. TAN, L. L. YONG, S. Z. SAHIBULDDIN, S. H. MAT DESA, N. A. WAHAB, R. HOD and H. J. TAN, *Prevalence and predictors of vitamin D deficiency among adults with epilepsy: a cross-sectional study*, Epilepsy and Behavior **147**, 2023, paper 109432.

[13] R. K. NARANG, G. G. GAMBLE, K. T. KHAW, C. A. CAMARGO, J. D. SLUYTER, R. K. R. SCRAGG and I. R. REID, *A prediction tool for vitamin D deficiency in New Zealand adults*, Archives of Osteoporosis **15**(172), 2020, pp. 1–11.

[14] R. G. CARRETERO, L. V. MEDINA, O. BARQUERO-PEREZ, I. MORA-JIMENEZ, C. SOGUERO-RUIZ and J. RAMOS-LOPEZ, *Machine learning approaches to constructing predictive models of vitamin D deficiency in a hypertensive population: a comparative study*, Informatics for Health and Social Care **46**(4), 2021, pp. 355–369.

[15] K. GONOODI, M. TAYEFI, M. SABERI-KARIMIAN, A. AMIRABADI ZADEH, S. DARROUDI, S. K. FARAHMAND, Z. ABASALTI, A. MOSLEM, M. NEMATY, G. A. FERNS, S. ESLAMI and M. G. MOBARHAN, *An assessment of the risk factors for vitamin D deficiency using a decision tree model*, Diabetes & Metabolic Syndrome: Clinical Research & Reviews **13**(3), pp. 1773–1777.

[16] M. KARAMIZADEH, M. SEIF, M. F. HOLICK and M. AKBARZADEH, *Developing a model for prediction of serum 25-hydroxyvitamin D level: the use of linear regression and machine learning methods*, Journal of the American Nutrition Association **41**(2), 2021, pp. 191–200.

[17] Z. AMIRI, M. NOSRATI, P. SHARIFAN, S. S. SOFLAEI, S. DARROUDI, H. GHAZIZADEH, M. M. BAJGIRAN, F. MOAFIAN, M. TAYEFI, E. HASANZADE, M. RAFIEE, G. A. FERNS, H. ESMAILY, M. AMINI and M. GHAYOUR-MOBARHAN, *Factors determining the serum 25-hydroxyvitamin D response to vitamin D supplementation: data mining approach*, BioFactors **47**(5), 2021, pp. 828–836.

[18] B. PADMAJA, B. R. REDDY, R. V. SAGAR, H. K. PRADHAN, G. C. SEKHAR and E. K. R. PATRO, *Prognosis of vitamin D deficiency severity using SMOTE optimized machine learning models*, Turkish Journal of Computer and Mathematics Education **12**(6), 2021, pp. 4553–4567.

[19] S. GUO, R. M. LUCAS, A. L. PONSONBY, C. CHAPMAN, A. COULTHARD, K. DEAR, T. DWYER, T. KILPATRICK, T. MCMICHAEL, M. P. PENDER, B. TAYLOR, P. VALERY, I. VAN DER MEI and D. WILLIAMS, *A novel approach for prediction of vitamin D status using support vector regression*, PLoS ONE **8**(11), 2013, paper e79970.

[20] S. BECHROUI, A. MONIR, H. MRAOUI, E. H. SEBBAR, E. SAALAOUI and M. CHOUKRI, *Predictive Analytics for Determining Patients' Vitamin D Status*, Studies in Big Data, First Ed., **53**, Springer, Cham, 2019.

[21] U. E. ESSIZ, O. H. YÜREGIR and E. SARAÇ, *Applying data mining techniques to predict vitamin D deficiency in diabetic patients*, Health Informatics Journal **29**(4), 2023, pp. 1–16.

[22] U. E. ESSIZ, Ç. İ. ACI, E. SARAÇ and M. ACI, *Supplementary material of the paper "Deep Learning-Based Prediction Models for the Detection of Vitamin D Deficiency and 25-Hydroxyvitamin D Levels Using Complete Blood Count Tests"*, Romanian Journal of Information Science and Technology, 2024. Accessed: September 24, 2024. [Online]. Available:https://drive.google.com/file/d/1zgpH7e7Ck7s8STBhi0QmoGpiobFhpGr/view?usp=sharing

[23] S. V. BUUREN and K. GROOTHUIS-OUDSHOORN, *MICE: multivariate imputation by chained equations in R*, Journal of Statistical Software **8**, 2011, pp. 1–67.

[24] Z. ZHENG, Y. CAI and Y. LI, *Oversampling method for imbalanced classification*, Computing and Informatics **34**(5), 2016, pp. 1017–1037.

[25] Y. XIE, C. ZHU, R. HU and Z. ZHU, *A coarse-to-fine approach for intelligent logging lithology identification with extremely randomized trees*, Mathematical Geosciences **53**, 2021, pp. 859–876.

[26] X. PANG and Y. XU, *A reconstructed feasible solution-based safe feature elimination rule for expediting multi-task lasso*, Information Sciences **642**, 2023, paper 119142.

[27] S. MORADI, *Real-time crash risk analysis using deep learning*, M.Sc. thesis, Department of Civil Engineering, İstanbul Technical University, İstanbul, Turkey, 2022.

[28] I. GOODFELLOW, Y. BENGIO and A. COURVILLE, *Deep Learning*, The MIT Press, Cambridge, MA, USA, 2016.

[29] Y. LECUN, L. BOTTOU, Y. BENGIO and P. HAFFNER, *Gradient-based learning applied to document recognition*, Proceedings of the IEEE **86**(11), 1998, pp. 2278–2324.

[30] H. KARAYIĞIT, Ç İNAN ACI and A. AKDAĞLI, *Detecting abusive Instagram comments in Turkish using convolutional neural network and machine learning methods*, Expert Systems with Applications **174**, 2021, paper 114802.

[31] Y. A. KADHIM, *Medical dataset classification based on different deep learning techniques and meta-heuristic algorithms*, Ph.D. thesis, Department of Electrical Electronics Engineering, Atılım University, İstanbul, Turkey, 2023.

[32] B. SU and S. LU, *Accurate recognition of words in scenes without character segmentation using recurrent neural network*, Pattern Recognition **63**, 2017, pp. 397–405.

[33] Y. BAI, J. XIE, C. LIU, Y. TAO, B. ZENG and C. LI, *Regression modeling for enterprise electricity consumption: A comparison of recurrent neural network and its variants*, International Journal of Electrical Power & Energy Systems **126**, 2021, paper 106612.

[34] S. GHIMIRE, Z. M. YASEEN, A. A. FAROOQUE, R. C. DEO, J. ZHANG and X. TAO, *Streamflow prediction using an integrated methodology based on convolutional neural network and long short-term memory networks*, Nature Scientific Reports **11**(1), 2021, pp. 1–26.

[35] R. DEY and F. M. SALEMT, *Gate-variants of gated recurrent unit (GRU) neural networks*, 2017 IEEE 60^{th} international Midwest symposium on circuits and systems, Boston, USA, 2017, pp. 1597–1600.

[36] L. E. E. OCHOA, I. B. R. QUINDE, J. P. C. SUMBA, A. V. GUEVARA and R. M. MENENDEZ, *New approach based on autoencoders to monitor the tool wear condition in HSM*, IFAC-PapersOnLine **52**, 2019, pp. 206–211.

[37] E. BISONG, *Python*, in Building Machine Learning and Deep Learning Models on Google Cloud Platform: A Comprehensive Guide for Beginners, 2019, pp. 71–89.

[38] B. PANG, E. NIJKAMP and Y. N. WU, *Deep learning with tensorflow: a review*, Journal of Educational and Behavioral Statistics **45**(2), 2019, pp. 227–248.

[39] E. BISONG, *Google Colaboratory*, in Building Machine Learning and Deep Learning Models on Google Cloud Platform: A Comprehensive Guide for Beginners, 2019, pp. 59–64.

[40] S. KAUR, H. AGGARWAL and R. RANI, *Hyper-parameter optimization of deep learning model for prediction of Parkinson's disease*, Machine Vision and Applications **31**, 2020, pp. 1–15.

[41] I. CINAR and M. KOKLU, *Classification of rice varieties using artificial intelligence methods*, International Journal of Intelligent Systems and Applications in Engineering **7**(3), 2019, pp. 188–194.