AI-Based Prediction for Glucose Levels: A Comparative Study of Machine Learning and Deep Learning Approaches

Amani Othmane*, Imane Youkana, Laid Kahloul, and Samir Bourekkache

LINFI Laboratory, Department of Computer Science, University of Biskra, Algeria
Othmaneamani00@gmail.com
{imane.youkana,l.kahloul,s.bourekkache}@univ-biskra.dz
* Corresponding author

Abstract. For the sake of diabetes management, patients need to measure their blood glucose level consistently , which could be challenging and stressful, since most of the time the used method is going to be invasive method which involves pricking the skin to obtain a blood sample to use it to measure the glucose level, or a semi-invasive method that requires the insertion of a sensor beneath the skin, so it still involve level of invasiveness. For this purpose, several studies have been done to accomplish the non invasive measurement that can make people with diabetes more relaxed while checking their blood glucose level daily, using different machine learning and deep learning algorithms, different sensors and devices and different physiological factors, such as Photoplethysmography (PPG). In this paper, we compare using numerous metrics the performance of Machine Learning and Deep Learning techniques, in order to predict glucose levels non invasively using our collected PPG dataset.
We assess the performance of several machine learning algorithms including Linear Regression, Lasso regression, Ridge Regression, Support Vector Regression, Gradient Boosting, Regressor, AdaBoost Regressor, XGB Regressor, and also, different deep learning algorithms CNN, RNN, CLDNN. All the experiments were conducted using our collected dataset. The results show that the Convolutional Neural Network (CNN) outperformed significantly the rest of models. Where, it provides the lowest values for most error metrics (MAE, MSE, RMSE, MAPE), a very high R2 score, indicating superior performance in minimizing prediction errors.

Keywords: Blood glucose Prediction · Photoplethysmography (PPG) · Machine learning · Deep learning · Regression · CNN.

1 Introduction

The diabetes mellitus disease is a chronic metabolic disorder [1] occurs when the blood sugar levels are elevated that result from either when the body doesn't produce enough insulin (type 1 diabetes) [2] or a resistance to insulin's effects in

C. A. Kerrache et al. (eds.), *Proceedings of the International Conference on Emerging Intelligent Systems for Sustainable Development (ICEIS 2024)*, Advances in Intelligent Systems Research 184,
https://doi.org/10.2991/978-94-6463-496-9_18

the body's cells (type 2 diabetes) [3]. Numerous factors influence and cause diabetes including genetic risk factors, lifestyle factors such as poor nutrition and lack of physical activity, obesity. These are all major contributors to the alarming grown of people with diabetes [3]. To prevent the potential complications of this disease and enhance patients' well-being and reduce healthcare costs associated with the condition, strict monitoring and management are crucial [4]. Traditionally, blood glucose levels are measured through invasive methods, such pricking the finger, which can be painful, uncomfortable, leads to discomfort about regular task and carries the risk of infection [5]. Additionally for pediatric patients, the discomfort may instill fear, causing them to avoid blood glucose testing and resist medical treatment. Thus, it is clear that developing a non-invasive blood sugar monitoring method is necessary.

Photoplethysmography (PPG) is an optical technique that uses light to measure blood volume changes non invasively [6]. It operates by emitting light into the skin and measures variations in light absorption using a photo-detector [7]. Several prediction based techniques were investigated for the non invasive blood glucose level prediction using PPG, showing promising results [8,9,10]. The core approach is to analyze the PPG data, relevant features can be extracted from the signal that might correlate with blood glucose levels such as systolic and diastolic peak times, oxygen saturation and heart rate as these might be influenced by blood sugar concentration.

To establish the relationship between PPG data and glucose levels, leveraging AI algorithms opens doors to predicting blood glucose levels accurately. Artificial intelligence (AI) is transforming the healthcare industry lately rapidly, several studies across various healthcare domains have been conducted to explore the applications of AI for improving patient care, remarkable progress is witnessed in areas including Disease Detection and Classification where AI techniques like deep learning are proving valuable for tasks such as fracture detection and classification [11,13], and diabetic detection in eye images [12] which is relevant to our work. Prediction and Monitoring where AI algorithms can analyze existing data to predict future values. For example, AI can predict the future vital signs based on existing data [14], or estimate age from medical images [15]. These studies highlights the potential of AI in healthcare, particularly for tasks involving prediction and analysis. In the context of non-invasive blood glucose measurement AI-based techniques also showcased significant promise in predicting glucose levels. Numerous machine learning algorithms have been implemented for glucose level prediction such as Support Vector Machine [16,17], Random Forest Regression [18,19], and K-Nearest Neighbors [20].
In addition, deep learning approaches as Deep Neural Networks (DNNs) [21,24,22], Convolutional Neural Networks (CNNs) [23,24], have shown many promising results in the prediction of glucose levels [25].

In this work, we present a comparative study to assess and compare the performance of AI algorithms in predicting blood glucose levels using PPG data. We investigate the performance of several machine learning algorithms such as Linear Regression, Lasso regression, Ridge Regression, Support Vector Regression, Gradient Boosting, Regressor, AdaBoost Regressor, XGB Regressor, and also, different deep learning algorithms CNN, RNN, CLDNN on our collected dataset using various metrics.

The paper plan is organized as follows: Section 2 presents the study methodology, next in Section 3 we provide a comparison between he results. Finally, we conclude and present the future works in last section.

2 Methodology

Our proposed system consists of several key stages, that are combined to fulfill the research goals, which is the blood glucose levels prediction as illustrated in Fig. 1. Using the ppg sensor, We collect the dataset and pre-processes it. Next, the pre-processed data is divided into training and testing sets.

Then, the data is fed to the proposed models for training, to predict glucose levels, Finally, the performance of these algorithms is evaluated using a comparative study on the unseen test set, and finally, the comparative study of the performance is evaluated using evaluation metrics. These steps will be briefly discussed in the following section.

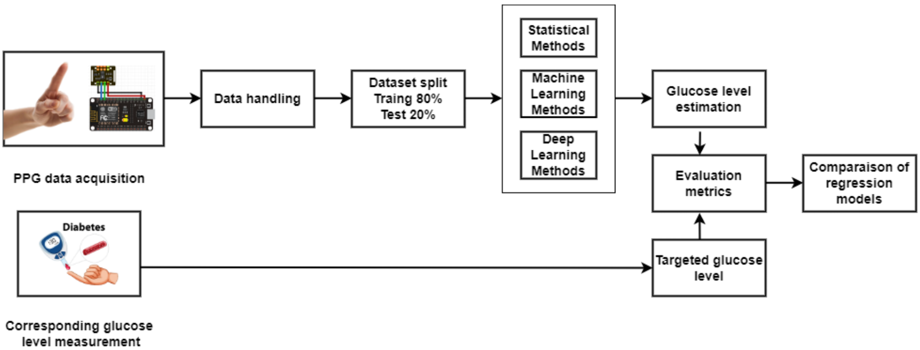

Fig. 1. Graphical illustration of our proposed methodology.

2.1 Data Collection and Description

We constructed an Algerian population-specific dataset, To fulfill the research objective Where, fifteen (15) volunteers took part in the experimentation, between the ages of 19–60 years, of which six subjects were female, and the rest were males. After a two-hour fast, from each subject, Red channel and IR channel PPG data, and corresponding glucose level through invasive measurements(using a VitalCheck glucometer). To assess changes over time, data was collected from each subject twice. Each data collection period lasted two minutes and maintained a steady state. The measurements were separated by a thirty-minute interval. We have used the same Glucometer and the same device for acquiring the blood glucose level of all the subjects.

Despite the small size of our dataset, several studies have demonstrated the efficacy of AI models in predicting blood glucose levels using similarly small datasets such as [27] that used a small dataset of 23 participants to train a CNN model for non-invasive glucose monitoring, achieving significant improvements in accuracy by employing an oversampling technique to balance the data. In addition, [28] collected PPG signals from 30 participants to measure blood glucose levels.

2.2 Pre-Processing

In order to improve computational efficiency and streamline the dataset for further analysis, we downsampled the data by averaging these values into 12 sets. Then, we transform nominal attribute values to numeric values, 2 for male and 1 for female. Also, we standardize the continuous features (red light, infrared light, and age) using StandardScaler to ensure all features are on a similar scale. After, the pre processed we split the data into 80% for training and 20% for testing.

2.3 Models

1. **Statistical Methods**

   - **Linear Regression:** is a technique that estimates the connection between various features β and a continuous target variable represented by y and several independent variables (x_1 to x_P) using a straight line through a process called least squares regression . It applies a regression equation to predict runoff based on rainfall data aiding in understanding the impact of rainfall variations as shown in the following Equation 1 [29]:

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + ... + \beta_p x_p + \varepsilon \tag{1}$$

   - **Lasso Regression:** also known as L1 regularization, it starts with the standard linear regression model. In addition, an absolute value penalty on the coefficients [30].

$$L\_1 = \lambda \times \beta_0 + \beta_1 x_1 + \beta_2 x_2 + ... + \beta_p x_p + \varepsilon \tag{2}$$

   - **Ridge Regression:** known as $L2$ regularization, is an adaptation of linear regression, used for analyzing multiple regression data that suffer from multicollinearity [31].

$$L\_2 = \lambda \times (\beta_0^2 + \beta_1^2 + \beta_2^2 + ... + \beta_p^2) \tag{3}$$

2. **Machine Learning Methods**

- **Support Vector Regression (SVR):** Unlike linear regression, SVR focuses on finding the max fit line is the hyperplane that finds the margin between the plane and the data points that is closer the most, and it is a type of Support Vector Machine (SVM) but it is used for continuous values, [32].
- **Gradient Boosting Regressor:** it combines several weak learners into a sinle more strong and accurate models, with each one focusing on improving the errors of the previous ones [33].
- **AdaBoost Regressor:** is a meta-estimator, similar to Gradient Boosting, AdaBoost which is short for Adaptive Boosting. It works by combining multiple weak learners to create a strong classifier or regressor [34].
- **XGB Regressor:** it is the abbreviation of Extreme Gradient Boosting which is an advanced implementation of Gradient Boosting. It contains a loss function and a regularization term [35].

3. **Deep Learning Models**

- **Convolutional Neural Network (CNN):** is a specialized type of artificial neural network. Unlike traditional neural networks, CNNs excel at automatically learning important features directly from the data through convolutional layers that act like filters, CNNs also uses pooling layers to reduce the data size, finally it include fully-connected layers similar to those found in other neural networks process the extracted features to make predictions or classifications [40]. And while CNNs are traditionally used for image recognition, their strength in extracting features can be applied for regression tasks. In our case, we used this feature extraction strength and implemented a CNN architecture for our regression problem.

  As illustrated in Fig. 2, our CNN architecture has several layers where the model starts with a 64-filter, 3×3 convolutional layer, 2×2 MaxPooling layer is included followed by a batch normalization layer to improve training stability and reduce overfitting. Three more convolutional layers are stacked 96, 208, and 64 as the numbers of filters respectively, a kernel size of 3 and the tanh activation function, maxPooling layers with a pool size of 3×3 are inserted after each convolutional layer. After that a flatten layer is included to transform the extracted features from a 3D tensor into a 1D vector to feed it to a single dense layer with one neuron and no activation function. At of 0.001 learning rate, the model was trained using the RMSprop optimizer.

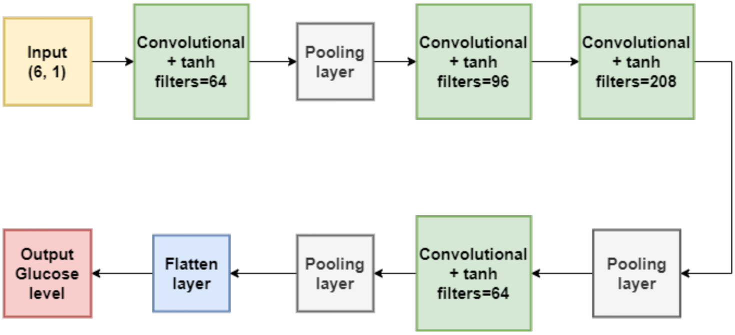

Fig. 2. CNN model architecture.

- **Convolutional Long Short-Term Memory Deep Neural Network (CLDNN):**
  In this network, the first layer is a convolutional layer with 64 filters of size 3×3, which is used for extracting features using the ReLU activation function. The result become the input to the MaxPooling layer with a pool size of 2, reducing its dimensionality and computational cost. Batch normalization is included similar to the previous CNN model. The pooled feature maps are input into two LSTM layers to capture temporal dependencies within the PPG signal.
  The first LSTM layer has 64 neurons and uses ReLU activation, while the second layer has 32 neurons and employs tanh activation for further temporal information processing.
  A fully-connected layer with ReLU activation with 32 neurons integrates the extracted features from both the CNN and LSTM layers. A final dense layer with a single neuron and no activation function.
  The model is trained using the RMSprop optimizer, learning rate of 0.001 and the mean squared error (MSE) loss function.
- ***Recurrent Neural Network (RNN):*** The network starts with a recurrent layer, specifically a SimpleRNN with 64 neuronshis layer utilizes a ReLU activation function to process the input data and capture temporal dependencies. A Batch Normalization layer is applied after the RNN layer. The normalized output is then fed into a fully-connected layer with a ReLU activation function and 32 neurons to integrate the extracted features from the RNN layer, finaly its fed to a dense layer with a single neuron and no activation function.

2.4 Performance evaluation Metrics

- **Mean Absolute Error (MAE):** It measures the median of the differences between the target and the predicted as shown in Equation 4 [36].

$$\frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i| \tag{4}$$

- **Mean Squared Error (MSE):** it represents the median of the squared differences between target and predicted as shown in Equation 5 [36].

$$\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2 \tag{5}$$

- **Root Mean Squared Error (RMSE):** It is the square root of the MSE, see Equation 6 [36].

$$\sqrt{MSE} \tag{6}$$

- **R-squared (R²) Score:** It represents the proportion of variance in the dependent variable explained by the model as shown in Equation 7. Here, $\overline{y}$ represents the average of the actual values [37].

$$1 - \frac{\sum_{i=1}^{n}(y_i - \overline{y})^2}{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2} \tag{7}$$

- **Mean Absolute Percentage Error (MAPE):** It computes the average magnitude of the percentage difference between the targeted and actual values as presented in Equation 8 [36,38].

$$\frac{1}{n}\sum_{i=1}^{n}\left|\frac{y_i - \hat{y}_i}{y_i}\right| \times 100\% \tag{8}$$

- **Median Absolute Error (MedAE):** stands for Median Absolute Error (MAE),it is the median value of $|y_i - \hat{y}_i|$ [39].
- **Mean Squared Logarithmic Error (MSLE):** it focuses on errors in percentage terms, it is considered as the average of the squared differences between the logarithms of the targeted and actual values as shown in Equation 9 [36].

$$\frac{1}{n}\sum_{i=1}^{n}(\ln(y_i + 1) - \ln(\hat{y}_i + 1))^2 \tag{9}$$

3 Experimental Results

Building on the error metrics introduced earlier, this section, analyzes and compares the results obtained from various machine learning and deep learning models.

3.1 Statistical Methods

Table 1 summarizes the performance of the four Statistical regression methods using various metrics.

As seen in Table 1, Ridge Regression appears to consistently achieving the best results overall, which means the best performance in blood glucose levels prediction.

Table 1. Statistical Methods metrics comparison.

Metrics	Methods	Results
Mean Absolute Error (MAE)	Linear Regression	0.6819606428152505
	Lasso Regression	0.7887602989846738
	Ridge Regression	**0.6091908166508478**
Mean Squared Error (MSE)	Linear Regression	0.8937598227929222
	Lasso Regression	1.046999959304863
	Ridge Regression	**0.48336385448883756**
Root Mean Squared Error (RMSE)	Linear Regression	0.9453887151817089
	Lasso Regression	1.0232301594972966
	Ridge Regression	**0.695243737468262**
R-squared (R2) Score	**Linear Regression**	**0.3156539619155325**
	Lasso Regression	0.3197700777392293
	Ridge Regression	0.6941675538383888
Mean Absolute Percentage Error (MAPE)	Linear Regression	0.05892996848687178
	Lasso Regression	0.06491890092819569
	Ridge Regression	**0.05133890047629035**
Median Absolute Error (MedAE)	**Linear Regression**	**0.49921203464548025**
	Lasso Regression	0.6427232811049732
	Ridge Regression	0.6882333973445114

- **MAE, MSE, RMSE, MAPE:** Ridge Regression achieved the lowest values.
- **R-squared (R2) Score:** while Linear Regression has the highest R2 score, it's important to note that Ridge Regression still have a high R2.
- **Median Absolute Error (MedAE):** Ridge Regression still has a competitive MedAE value but not the best.

3.2 Machine Learning Methods

Table 2 summarizes the performance of the four machine learning regression methods using various metrics. Based on shown results, XGB Regressor appears to be the most effective method, with the lowest errors (MAE, MSE, RMSE, MAPE), with a significant R2 score and a low median absolute error (MedAE).

- **MAE, MSE, RMSE, MAPE:** XGB Regressor achieved the lowest values.
- **R-squared (R2) Score:** Both XGB Regressor and Gradient Boosting Regressor achieved high R2 scores. However, Support Vector Regression and AdaBoost Regressor also have very good R2 scores, indicating strong relationships between features and the target variable for all methods.
- **Median Absolute Error (MedAE):** Similar to the other error metrics, XGB Regressor again has the lowest MedAE.

Table 2. Machine Learning Methods metrics comparison.

Metrics	**Methods**	**Results**
Mean Absolute Error (MAE)	Support Vector Regression	0.10187714948462961
	Gradient Boosting Regressor	0.3415853812248867
	AdaBoost Regressor	0.08334203870841844
	XGB Regressor	**0.01257032506606158**
Mean Squared Error (MSE)	Support Vector Regression	0.028284827416281827
	Gradient Boosting Regressor	0.6728659855980343
	AdaBoost Regressor	0.01512163834706641
	XGB Regressor	**0.005215066366979633**
Root Mean Squared Error (RMSE)	Support Vector Regression	0.16818093654240907
	Gradient Boosting Regressor	0.8202840883486857
	AdaBoost Regressor	0.12297007094031624
	XGB Regressor	**0.07221541640799167**
R-squared (R2) Score	Support Vector Regression	0.9821343190654565
	Gradient Boosting Regressor	0.9917498427111366
	AdaBoost Regressor	0.9904486471866987
	XGB Regressor	**0.9999360569285409**
Mean Absolute Percentage Error (MAPE)	**Support Vector Regression**	**0.009254727257846818**
	Gradient Boosting Regressor	0.29092445117520843
	AdaBoost Regressor	0.7188335913380233
	XGB Regressor	0.012805862469419732
Median Absolute Error (MedAE)	Support Vector Regression	0.07989792882643965
	Gradient Boosting Regressor	0.019733025951623517
	AdaBoost Regressor	0.05769230769231015
	XGB Regressor	**0.000194549560546875**

3.3 Deep Learning Models

Table 3 summarizes the performance of three deep learning architectures using various metrics. Based on results of Table 3, CNN appears to be the most effective deep learning architecture, acheiving the lowest errors (MAE, MSE, RMSE, MAPE), and a high R2 score while having a low median absolute error (MedAE).

- ***MAE, MSE, RMSE, MAPE:*** CNN achieved the lowest values for MAE, MSE, RMSE, and MAPE
- ***R-squared (R2) Score:*** All three architectures achieved R2 scores, However, CNN has a slight edge over the others.
- ***Median Absolute Error (MedAE):*** Similar to the other error metrics, CNN again has the lowest MedAE.

Table 3. Deep Learning Methods metrics comparison.

Metrics	Methods	Results
Mean Absolute Error (MAE)	**CNN**	**0.047387266159057584**
	CLDNN	0.2694697660558364
	RNN	0.5183724796070772
Mean Squared Error (MSE)	**CNN**	**0.002879348923802342**
	CLDNN	0.07794856087660745
	RNN	0.5299751825719689
Root Mean Squared Error (RMSE)	**CNN**	**0.053659565072802645**
	CLDNN	0.2791926948840307
	RNN	0.7279939440489659
R-squared (R2) Score	CNN	0.9981813030564132
	CLDNN	**0.9990442556148815**
	RNN	0.9935018581574951
Mean Absolute Percentage Error (MAPE)	CNN	0.004082924482283074
	CLDNN	**0.0023507262025796**
	RNN	0.004879286557381349
Median Absolute Error (MedAE)	**CNN**	**0.03784732818603587**
	CLDNN	0.2885017395019531
	RNN	0.3443756103515625

3.4 Cross-Category Model Comparison

Table 4 illustrates the performance of the top models identified from each category using various metrics. This comparison aims to provide the best model for predicting glucose levels.

In summary, this cross-category comparison demonstrates that CNN provides the best overall performance in blood glucose levels predicting.

- **MAE, MSE, RMSE, MAPE:** CNN achieved the lowest values for all error metrics.
- **R-squared (R2) Score:** Both XGB Regressor and CNN achieved high R2 scores, Ridge Regression has a lower R2 score compared to these two methods.

- **Median Absolute Error (MedAE):** XGB Regressor achieved the absolute lowest MedAE value. However, CNN's MedAE is still very competitive.

Table 4. Cross-Category Model metrics comparison.

Metrics	**Methods**	**Results**
Mean Absolute Error (MAE)	CNN	0.047387266159057584
	XGB Regressor	**0.01257032506606158**
	Ridge Regression	0.6091908166508478
Mean Squared Error (MSE)	**CNN**	**0.002879348923802342**
	XGB Regressor	0.005215066366979633
	Ridge Regression	0.48336385448883756
Root Mean Squared Error (RMSE)	**CNN**	**0.053659565072802645**
	XGB Regressor	0.07221541640799167
	Ridge Regression	0.695243737468262
R-squared (R2) Score	CNN	0.9981813030564132
	XGB Regressor	**0.9999360569285409**
	Ridge Regression	0.6941675538383888
Mean Absolute Percentage Error (MAPE)	**CNN**	**0.004082924482283074**
	XGB Regressor	0.012805862469419732
	Ridge Regression	0.05133890047629035
Median Absolute Error (MedAE)	CNN	0.03784732818603587
	XGB Regressor	**0.00019454956054 6875**
	Ridge Regression	0.6882333973445114

4 Conclusion

In the purpose of measuring blood glucose levels non invasive using Photoplethysmography (PPG), we compared various AI models such as regression techniques which including statistical methods, machine learning and deep learning models. The analysis employed a range of metrics to assess the performance of the different implemented models such as the Mean Absolute Error (MAE), the Mean Squared Error (MSE), the Root Mean Squared Error (RMSE),the R-squared (R2) score, the Mean Absolute Percentage Error (MAPE), and the Median Absolute Error (MedAE).

In our experiments, Convolutional Neural Network (CNN) outperformed other machine learning and deep learning approaches. Where, it provides the lowest values for most error metrics that are MAE, MSE, RMSE and MAPE. An exceptionally high R2 score demonstrates superior performance in minimizing prediction errors.

The choice of the final model for real-world applications might extend beyond just raw performance metrics, XGB Regressor, for instance, achieved the lowest

MedAE, suggesting highly accurate median predictions. Furthermore, XGB Regressors are generally less complex and computationally expensive compared to CNNs, making them potentially more interpretable and easier to implement. In conclusion, our experimental results demonstrate the importance to consider various factors when selecting a model for glucose level prediction.

In future, we intend collecting additional data in order to build a more comprehensive dataset which will enhance the precision and accuracy of our glucose level prediction.

5 Acknowkedgment

This work is realized through the mixed team TIASM (Techniques de l'IA pour le Soutien de la Médecine) approved by ATRSS (Agence Thematique de la Recherche en Science de Santé) on March 17th 2021, under the agreement of the Ministry of high education and scientific research.

References

1. Kaul, K., Tarr, J.M., Ahmad, S.I., Kohner, E.M., Chibber, R.: Introduction to Diabetes Mellitus. In: Ahmad, S.I. (eds) Diabetes. Advances in Experimental Medicine and Biology, vol 771. Springer, New York, NY. `https://doi.org/10.1007/978-1-4614-5441-0_1`
2. Paschou, Stavroula A., et al. (2018). On type 1 diabetes mellitus pathogenesis. *Endocrine Connections*, **7**(1), R38-R46.
3. Mahler, Richard J., & Michael L. Adler. (1999). Type 2 diabetes mellitus: update on diagnosis, pathophysiology, and treatment. *The Journal of Clinical Endocrinology & Metabolism*, **84**(4), 1165-1171.
4. Chaudhury, Arun, et al. (2017). Clinical review of antidiabetic drugs: implications for type 2 diabetes mellitus management. *Frontiers in Endocrinology*, **8**, 6.
5. Bolla, Anmole S., & Ronny Priefer. (2020). Blood glucose monitoring-an overview of current and future non-invasive devices. *Diabetes & Metabolic Syndrome: Clinical Research & Reviews*, **14**(5), 739-751.
6. Abay, Tomas Y., & Panicos A. Kyriacou. (2022). Photoplethysmography in oxygenation and blood volume measurements. In *Photoplethysmography*, 147-188. Academic Press.
7. Kavya, R., et al. (2020). Photoplethysmography—A modern approach and applications. In *2020 International Conference for Emerging Technology (INCET)*. IEEE.
8. Daarani, P., and A. Kavithamani: Blood glucose level monitoring by noninvasive method using near infrared sensor. *International Journal of Latest Trends in Engineering and Technology* (2017), 141-147.
9. Singha, Suvojit Kumar, and Mohiuddin Ahmad: Noninvasive heart rate and blood glucose level estimation using photoplethysmography. In: *2021 International Conference on Information and Communication Technology for Sustainable Development (ICICT4SD)*. IEEE, 2021.
10. Sarkar, Kamol, et al.: Design and implementation of a noninvasive blood glucose monitoring device. In: *2018 21st International Conference of Computer and Information Technology (ICCIT)*. IEEE, 2018.

11. Salem, T., Belaiche, L., Kahoul, M., Kahloul, L., & Belouaar, H. (2023). One-Stage Deep Learning Approach for the Detection of Fractures in Cervical Spine. In *2023 Third International Conference on Theoretical and Applicative Aspects of Computer Science (ICTAACS)* (pp. 1-5). Skikda, Algeria. doi: 10.1109/ICTAACS60400.2023.10449644.
12. Fellah, K.M., Tigane, S., & Kahloul, L. (2023). Diabetic Retinopathy Detection Using Deep Learning. In: Chikhi, S., Diaz-Descalzo, G., Amine, A., Chaoui, A., Saidouni, D.E., Kholladi, M.K. (Eds.), *Modelling and Implementation of Complex Systems. MISC 2022.* Lecture Notes in Networks and Systems, vol 593. Springer, Cham..
13. Medaramatla, Sai Charan, et al. (2024). Detection of Hand Bone Fractures in X-ray Images using Hybrid YOLO NAS. *IEEE Access.*
14. Title: Precision Unleashed: Enhancing Vital Sign Predictions Using Long-Short-Term Memory Networks
15. Zerari, A., Djedidi, O., Kahloul, L., Carlo, R., & Remadna, I. (2022). Paediatric Bone Age Assessment from Hand X-ray Using Deep Learning Approach. In: Senouci, M.R., Boulahia, S.Y., Benatia, M.A. (Eds.), *Advances in Computing Systems and Applications. CSA 2022.* Lecture Notes in Networks and Systems, vol 513. Springer, Cham.
16. Castro-Pimentel, Luis Alfredo, et al.: Three-wavelength PPG and support vector machine for non-invasive estimation of blood glucose. *Optical and Quantum Electronics* **55**(8), 708 (2023).
17. Prabha, Anju, et al.: Non-invasive diabetes mellitus detection system using machine learning techniques. In: *2021 11th International Conference on Cloud Computing, Data Science & Engineering (Confluence).* IEEE, 2021.
18. Islam, Tanvir Tazul, et al.: Blood glucose level regression for smartphone PPG signals using machine learning. *Applied Sciences* **11**(2), 618 (2021).
19. Georga, Eleni I., et al.: A predictive model of subcutaneous glucose concentration in type 1 diabetes based on random forests. In: *2012 Annual International Conference of the IEEE Engineering in Medicine and Biology Society.* IEEE, 2012.
20. Altman, Naomi S.: An introduction to kernel and nearest-neighbor nonparametric regression. *The American Statistician* **46**(3), 175-185 (1992).
21. Gorochna, M., R. Rekha, and S. Baliga: Implementation of non-invasive blood glucose monitoring system. *Int. J. Eng. Res. Electron. Commun. Eng* 9 (2022), 15-21.
22. Hossain, Shamima, et al.: Estimation of blood glucose from PPG signal using convolutional neural network. In: *2019 IEEE International Conference on Biomedical Engineering, Computer and Information Technology for Health (BECITHCON).* IEEE, 2019.
23. Mahmud, Tasfin, et al.: Non-invasive blood glucose estimation using multi-sensor based portable and wearable system. In: *2019 IEEE Global Humanitarian Technology Conference (GHTC).* IEEE, 2019.
24. Schrumpf, Fabian, et al.: Assessment of non-invasive blood pressure prediction from PPG and rPPG signals using deep learning. *Sensors* **21**(18), 6022 (2021).
25. Refat, Md Abu Rumman and Al Amin, Md and Kaushal, Chetna and Yeasmin, Mst Nilufa and Islam, Md Khairul: A comparative analysis of early stage diabetes prediction using machine learning and deep learning approach, 2021 6th *International Conference on Signal Processing, Computing and Control (ISPCC)*, 654–659, 2021.
26. scikit-learn: Machine Learning in Python. `https://scikit-learn.org/stable/`

27. Putra, Karisma, et al. (2024). Optimizing the Performance of AI Model for Non-Invasive Continuous Glucose Monitoring: Hyperparameter Tuning and Random Oversampling Approach. *JOIV: International Journal on Informatics Visualization*, **8**(2), 991-1000.
28. Hossain, Shamima, et al. (2019). Estimation of blood glucose from PPG signal using convolutional neural network. *2019 IEEE International Conference on Biomedical Engineering, Computer and Information Technology for Health (BECITHCON).* IEEE, 2019.
29. Diskin, M. H.: Definition and uses of the linear regression model. *Water Resources Research* **6**(6), 1668–1673 (1970).
30. Hans, Chris: Bayesian lasso regression. *Biometrika* **96**(4), 835-845 (2009).
31. McDonald, Gary C.: Ridge regression. *Wiley Interdisciplinary Reviews: Computational Statistics* **1**(1), 93-100 (2009).
32. Smola, A. J., & Schölkopf, B. (2004). *A tutorial on support vector regression. Statistics and Computing*, **14**, 199–222. https://doi.org/10.1023/B:STCO.0000035301.49549.88
33. Keprate, Arvind, & RM Chandima Ratnayake. (2017). Using gradient boosting regressor to predict stress intensity factor of a crack propagating in small bore piping. In *2017 IEEE International Conference on Industrial Engineering and Engineering Management (IEEM).* IEEE.
34. Solomatine, Dimitri P., & Durga L. Shrestha. (2004). AdaBoost. RT: a boosting algorithm for regression problems. In *2004 IEEE International Joint Conference on Neural Networks (IEEE Cat. No. 04CH37541)* (Vol. 2). IEEE.
35. Arora, Mohit, et al. (2021). A State of the Art Regressor Model's comparison for Effort Estimation of Agile software. In *2021 2nd International Conference on Intelligent Engineering and Management (ICIEM).* IEEE.
36. Chai, Tianfeng, & Roland R. Draxler. (2014). Root mean square error (RMSE) or mean absolute error (MAE). *Geoscientific Model Development Discussions*, **7**(1), 1525-1534.
37. Lewis-Beck, Michael S., & Andrew Skalaban. (1990). The R-squared: Some straight talk. *Political Analysis*, **2**, 153-171.
38. De Myttenaere, Arnaud, et al. (2016). Mean absolute percentage error for regression models. *Neurocomputing*, **192**, 38-48.
39. Savini, Giacomo, et al. (2021). Comparison of 13 formulas for IOL power calculation with measurements from partial coherence interferometry. *British Journal of Ophthalmology*, **105**(4), 484-489.
40. Bouvrie, Jake: Notes on convolutional neural networks. (2006).