Machine Learning Applications in Diabetes Diagnosis: Optimizing Predictive Models

Byung Joo Kim

Professor, Department of EE, Youngsan University, South Korea, bjkim@ysu.ac.kr

Millions of individuals globally suffer from diabetes, which is a persistent health condition. Proper management of diabetes is crucial to prevent complications and improve the quality of life for patients. This study aims to develop a diabetes management model using logistic regression and multi-layer perceptron (MLP) neural networks on the Pima Indian diabetes dataset. In this study the logistic regression technique is employed to estimate the probability of a patient having diabetes based on the patient's data, while MLP neural networks are a powerful method for modeling complex relationships between features and outcomes. According to experimental results, the precision, recall and F1 score are 0.8607, 0.8813 and 0.871 respectively. This suggests that the classifier has a high accuracy for positive predictions (precision) and a high coverage of positive examples (recall), leading to a high F1 score. These results indicate that the classifier is performing well in both correctly identifying positive cases and accurately identifying the proportion of positive cases in heavily skewed data. These findings demonstrate the potential of predictive modeling for diabetes management and contribute to the development of better tools for diabetes management. This study's major achievements can be summarized as follows. First diabetes prediction models enable early identification of individuals at high risk of developing diabetes. This early detection allows for timely intervention and preventive measures to be implemented, such as lifestyle modifications, dietary changes, and targeted interventions, to prevent or delay the onset of diabetes. Second predictive models of diabetes help healthcare professionals in assessing an individual's risk profile and tailoring personalized treatment plans. This can lead to better management of the disease, improved patient outcomes, and optimized allocation of healthcare resources. Third diabetes prediction research has contributed to the advancement of machine learning and data science techniques. These models have helped refine algorithms, feature selection methodologies, and model evaluation techniques. This can benefit other areas of healthcare and biomedical research where predictive modeling is applied.

Keywords: Diabetes Management, Logistic Regression, MLP, Risk Assessment, Predictive Model.

1. Introduction

Diabetes affects millions of people across the world, where the body's inability to produce or use insulin adequately results in elevated blood glucose levels. If not treated, diabetes can result to severe consequences such as kidney failure, blindness, and cardiovascular disease. Thus, it is essential to identify and manage diabetes early to avoid these complications. With around 13% of the population affected by the ailment, Korea is among the countries with the highest diabetes prevalence globally. In recent years, there has been a notable rise in the incidence of diabetes cases in the country, due to a variety of factors, including an aging population, urbanization, and changes in lifestyle and dietary habits. The Korean government has implemented various policies and programs to address the prevalence of diabetes. These include nationwide screenings for diabetes, public education campaigns on diabetes prevention and management, and the development of national guidelines for diabetes care. Despite these efforts, there are still challenges in managing diabetes in Korea. Diabetes prevention is important in Korea because it helps reduce the risk of developing the disease and its complications. This can be achieved through early screening and management of risk factors[1]. However, early prediction of diabetes is quite a challenging task for medical practitioners [2]. Rather than relying on long-term medication after an event has occurred, it is more beneficial to predict its occurrence earlier in order to prevent it. [3]. So, in this paper, a diabetes predicting system based on machine learning techniques, specifically logistic regression and a multilayer perceptron (MLP) was proposed. The logistic regression algorithm was used to predict the probability of a patient having diabetes based on the patient's data. The MLP was then used to classify the patients into one of three categories: normal, prediabetes, or diabetes. These algorithms were trained using a dataset of patient data, including demographic information, laboratory results, and medical history. This study's major achievements can be summarized as follows. First, diabetes prediction models enable early identification of individuals at high risk of developing diabetes. This early detection allows for timely intervention and preventive measures to be implemented, such as lifestyle modifications, dietary changes, and targeted interventions, to prevent or delay the onset of diabetes. Second, predictive models of diabetes help healthcare professionals in assessing an individual's risk profile and tailoring personalized treatment plans. This can lead to better management of the disease, improved patient outcomes, and optimized allocation of healthcare resources. Third, diabetes prediction research has contributed to the advancement of machine learning and data science techniques. These models have helped refine algorithms, feature selection methodologies, and model evaluation techniques. This, in turn, can benefit other areas of healthcare and biomedical research where predictive modeling is applied. This paper is composed of following sections. Chapter 2 provides theoretical justifications for the use of logistic regression and neural networks. Chapter 3 explains the data, proposed model, and the experimental results based on it. Chapter 4 presents the analysis of the experimental results and suggestions for future research directions.

2. Theoretical Background of Proposed Model

2.1 Logistic Regression

In this paper, logistic regressions utilized[4]. There are several reasons why logistic regression was used in this study. First, logistic regression models can estimate the probability or risk of developing diabetes based on a set of predictor variables. These models can assist in identifying individuals who have a higher likelihood of developing diabetes and prioritize preventive interventions. Risk assessment is crucial for targeted screening, early detection, and implementing appropriate preventive strategies. Second, logistic regression provides easily interpretable results. The coefficients associated with each predictor variable indicate the direction and magnitude of their influence on the probability of diabetes. This interpretability is valuable for clinicians and researchers in understanding the risk factors and making informed decisions regarding diabetes management. Third, logistic regression is a computationally efficient algorithm, especially compared to more complex machine learning models. It can handle large datasets with many predictor variables and is less prone to overfitting. This efficiency makes logistic regression an attractive choice for analyzing diabetes-related data, which often involves a wide range of clinical, demographic, and lifestyle factors. Fourth, logistic regression models can be integrated into clinical decision support systems, providing valuable assistance to healthcare professionals in managing diabetes. These models can assist in risk stratification, treatment planning, and monitoring the progress of patients. Logistic regression provides a transparent and interpretable framework to support clinical decision-making. While logistic regression has its limitations, such as assuming a linear relationship between predictors and the log odds of diabetes, its interpretability, efficiency, risk assessment capabilities, and clinical decision support have made it a reasonable and commonly used approach in diabetes management. Before presenting the proposed model, a brief explanation is given on logistic regression. The logistic regression method is utilized for addressing binary classification problems, these models aim to predict a binary outcome by considering one or more input variables. The technique employs a logistic function, commonly referred to as the sigmoid function, is employed to model the relationship between the input variables and the binary outcome. The sigmoid function transforms any real-numbered value into a value ranging between 0 and 1, which can be construed as a probability. The logistic regression model can be represented by the following equation.

$$P(y = 1|x|) = \frac{1}{\left(1 + e^{\left(-(b_0 + b_1 x_1 + b_2 x_2 + \dots + b_n x_n)\right)}\right)}$$

where$P(y = 1|x|)$ is the probability that $y = 1$ given the input variables x, e is the base of the natural logarithm, $b_0, b_1, b_2 \dots, b_n$ are the parameters (or coefficients) of the model, and $x_1, x_2, x_3, \dots, x_n$ are the input variables. Maximum likelihood estimation is a technique used to estimate the parameters of the logistic regression model by identifying the values that maximize the likelihood of the observed data. After the parameters are determined, the model can make predictions about new, unobserved data. Logistic regression is a straightforward and interpretable algorithm that is suitable for binary classification problems. It is easy to implement, computationally efficient, and does not require a large amount of data. However, it has some limitations, such as the assumption of linearity between the input variables and the log odds of the outcome, and the assumption that the observations are independent.

2.2 Multilayer Perceptron(MLP)

From a theoretical point of view, an ensemble method combines multiple classifiers to make predictions, which can often lead to improved performance compared to a single classifier. Ensemble methods, such as random forest or voting classifiers, can help reduce bias and variance, enhance the robustness of predictions, and handle complex relationships in the data[5][6]. However, there are a few points to consider when choosing the classifier between ensemble methods and single classifiers. The main point we consider is computational complexity. Ensemble methods, particularly those that combine multiple models, can be computationally more expensive compared to single classifiers[7][8]. This is because they require training and combining multiple models. If computational resources are limited, a single classifier may be preferred. The choice between an ensemble method and a single classifier depends on various factors such as the dataset characteristics, computational resources, interpretability requirements, and the specific problem at hand. It is important to consider the trade-offs and choose the method that best suits the specific needs of the application. We compared the performance of ensemble methods and single classifiers on the Pima Indian dataset. The experimental results are shown in Table 1.

Table 1. Performance comparison between ensemble method and single classifier

Method	Accuracy	Precision	Recall	F1 Score
Ensemble	0.766	0.693	0.618	0.653
SVM	0.766	0.72	0.563	0.632
MLP	0.751	0.692	0.626	0. 647
Decision Tree	0.746	0.625	0.657	0.645
Random Forest	0.746	0.648	0.636	0.642

As shown in Table 1, in terms of accuracy, the ensemble method and SVM with RBF kernel function slightly outperformed MLP, and Decision tree, but they typically require more computational resources and memory. In terms of the F1 score, the ensemble method showed slight superiority, followed by the MLP method. Therefore, in this paper, the MLP method was used as the classifier, considering its performance in terms of both accuracy and F1 score, while also taking into account computational complexity and memory requirements. Before presenting the proposed model, a brief explanation is given on MLP. MLP, a form of neural network, is a versatile tool that can be applied to various machine learning tasks such as regression and classification. MLP is composed of one or more layers of artificial neurons, also known as Perceptrons, that are connected by directed edges, or weights. The layers are typically organized in a feedforward structure where the output of one layer is the input to the next layer. The forward pass in a MLP is the process of calculating the output of the network given an input. It is also known as the feedforward process. The forward pass starts with the input vector x, which is passed through the first layer of the network. In the first layer, the input vector is multiplied by the weight matrix w^1 and added to the bias term. The result is then passed through the activation function, $\sigma(\quad)$, which produces the output of the first layer. This output is then used as the input for the next layer where the same process is repeated. The input for the next layer is the output of the previous layer, multiplied by the next weight matrix w^2 and added to the bias term. The result is passed through the activation function again. This process is repeated for all layers of the network. In the final layer, the output of the previous layer is multiplied by the weight matrix w^n and added to the bias term. The final output y is calculated by applying the activation function on the resulting vector. Figure 1 illustrates the forward pass process in a MLP. It is important to note that the same activation function is used throughout all layers or different activation functions can be applied in different layers depending on the problem.

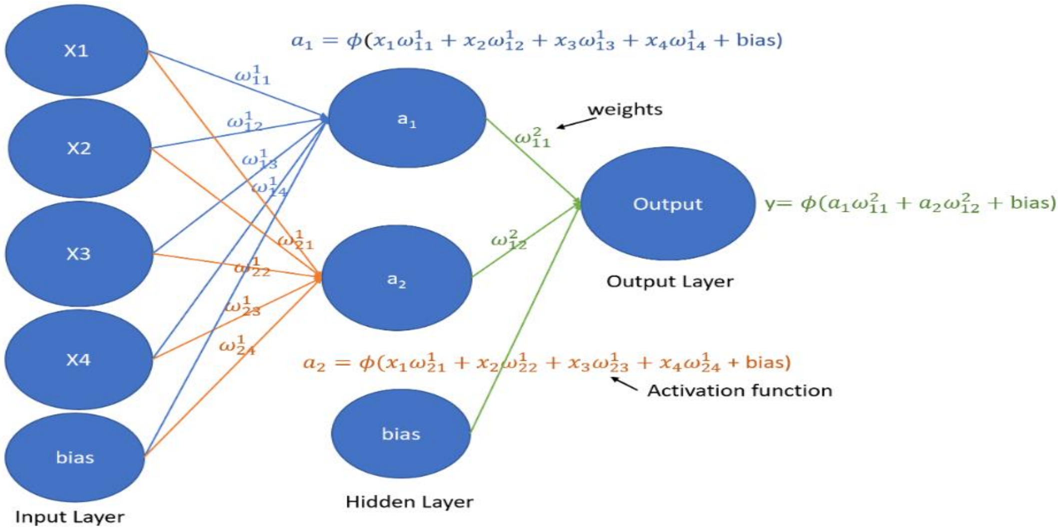

Figure 1. Multilayer perceptrom

In an MLP, The process of adjusting the weights and biases of the network to minimize the error between the predicted output and the actual output is known as backpropagation or backward pass. The backward pass starts with the final output y, which is compared to the true output to calculate the error. In a MLP, the backward pass refers to the process of updating the weights and biases of the network based on the calculated gradients during backpropagation. overview of the backward pass of an MLP is as follows. First, calculate the gradient of the loss function with respect to the output layer activations using the selected loss function and the predicted output of the MLP. Second, propagate the gradients backward through the network, layer by layer, to calculate the gradients of the loss function with respect to the weights and biases of each layer. To accomplish this, the chain rule of calculus is applied. Afterward, the weights and biases of each layer are updated using an optimization algorithm like gradient descent or its variations. This process involves adjusting the weights and biases in the opposite direction of the gradients, with the learning rate determining the magnitude of the updates. The first to third steps are repeated for a certain number of epochs or until a convergence criterion is met. The backward pass is a crucial step in training an MLP as it allows the network to learn and adjust its parameters to minimize the loss function and improve its performance on the given task. The gradient of the weight matrix and bias term in the last layer is computed with respect to the error and used to update these parameters using an optimization algorithm, such as gradient descent. This procedure is then repeated for all layers in the network, with the error being propagated back and the weight matrix and bias term being updated in each layer based on their respective gradients.

3. Proposed Model

The proposed model has the components as follows. The logistic regression algorithm is used to estimate the probability of a patient having diabetes based on the patient's data. Based on the calculated probability, MLP is then used to classify the patients into one of three categories: normal, prediabetes, or diabetes.

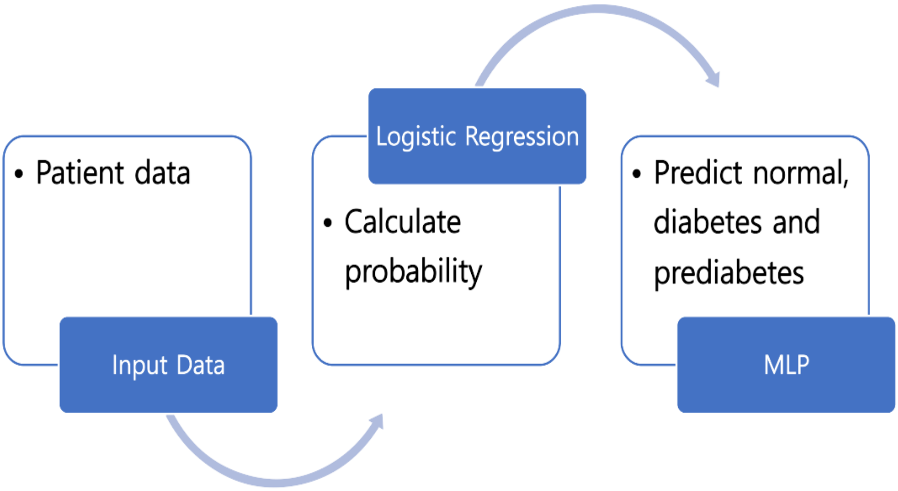

Figure 2. Proposed Model

To predict whether a patient is normal, prediabetic, or diabetic, the output of the classification algorithm needs to be transformed into three categories. This can be done using different threshold values for the predicted probability. In this model, a threshold value of 0.5 can be used to divide the patients into normal (probability < 0.5) and diabetic (probability >= 0.5). The threshold for distinguishing between normal and prediabetes is set to 0.3. If the predicted probability of a patient having diabetes is greater than 0.3, they will be classified as prediabetic. If the predicted probability is less than 0.3, they will be classified as normal. The threshold value of 0.3 for prediabetes classification has not yet been supported by any medical evidence. In the future, if a more accurate threshold value for prediabetes is provided, it should be incorporated.

3.1 Data

Unfortunately, in this paper, the researchers were unable to obtain data for Korean diabetes patients, instead planned to use the Pima Indian Diabetes dataset [9]. The Pima Indian Diabetes dataset is a commonly used dataset for the prediction of diabetes, and it can provide valuable insights. The Pima Indian Diabetes dataset is used for predictive modeling in healthcare. It contains information on various health measurements for 768 patients, including the following 8 features for each patient. Pregnancies, glucose, blood pressure, skin thickness (triceps skin fold thickness), insulin(2-hour serum insulin), BMI, pedigree, and age. The target label for each patient is binary, indicating whether the patient has diabetes (1) or not (0). Table 2 provides information about the data types and descriptions of the Pima Indian data.

Table 2. information about the data types and descriptions of the Pima Indian data.

Features	Data Type	Description
Pregnancies	Numeric(integer)	Number of times pregnant
glucose	Numeric(integer)	Plasma glucose concentration
blood pressure	Numeric(integer)	Diastolic blood pressure
skin thickness	Numeric(integer)	triceps skin fold thickness
insulin(2-hour serum insulin)	Numeric(integer)	2-hour serum insulin
BMI	Numeric(float)	Body mass index
pedigree	Numeric(float)	Diabetes pedigree
age	Numeric(integer)	Age (years)
diabetes	Numeric	Target value(diabetes 1 , normal 0)

3.2 EDA (Exploratory Data Analysis)

EDA is an essential step in the data analysis process. It involves thoroughly examining and understanding the data in order to gain insights, identify patterns, and detect anomalies. EDA techniques help in uncovering relationships between variables, understanding the distribution of data, and assessing data quality[10].

(1) Distribution

In the Pima Indian dataset, the ratio of those with diabetes to those without diabetes is 65% and 35%, respectively. And the ratio of individuals with diabetes, individuals with prediabetes, and individuals without diabetes, as defined by the research, is 52%, 20%, and 27% respectively.

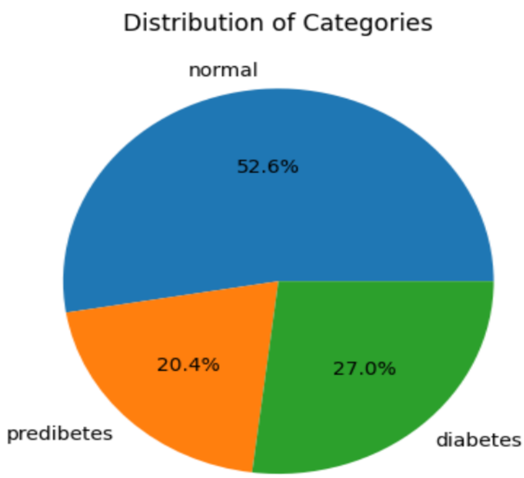

Figure 3. Distribution of Pima Indian Data

(2) Feature Ranking

Feature ranking refers to the process of determining the importance or relevance of each feature in a dataset for a particular task, such as classification or regression. In feature ranking, the correlation between the features in a dataset and their relationship to the target variable is assessed. This analysis helps identify the most important features that are strongly correlated with the target variable, which can be valuable for understanding the predictive power of each feature. There are several methods to perform feature ranking. First is Univariate Feature Selection. This method evaluates each feature independently based on statistical tests, such as chi-square, ANOVA, or correlation, and selects the most relevant features. Second is Recursive Feature Elimination (RFE). RFE begins with all the features and iteratively eliminates the least significant ones based on a model's performance. It continues until a specified number of features is reached. Third is Importance from Tree-based Models. Decision tree-based models like Random Forest or Gradient Boosting are capable of providing a feature importance score, which indicates the contribution of each feature towards the model's predictive performance. The last one is L1 Regularization (Lasso). L1 regularization can be used to penalize the coefficients of less important features, effectively setting them to zero. The features with non-zero coefficients are considered the most important. Once the features are ranked, they can be used to select the top-k features or to discard the least important ones. Feature ranking helps to reduce the dimensionality of the dataset, improve model performance, and gain insights into the underlying relationships between features and the target variable.

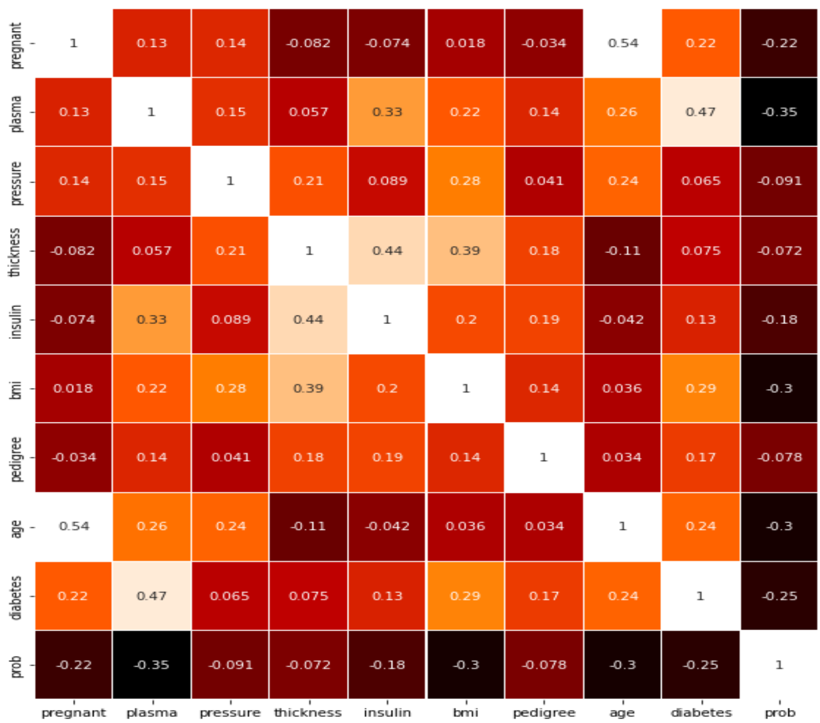

Figure 4. Correlation between features

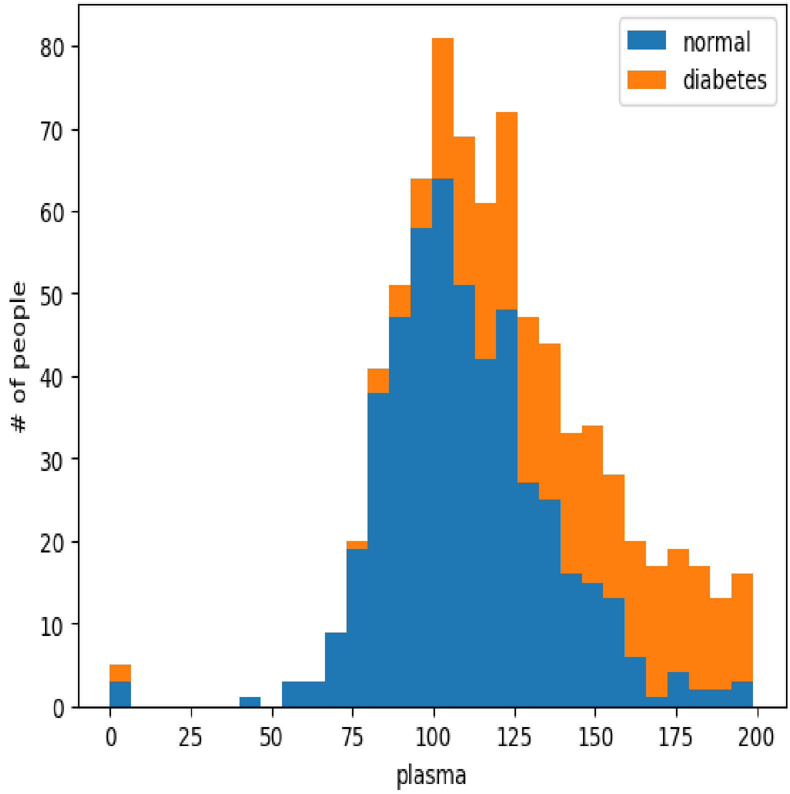

Figure 5. Distribution of people according to Plasma

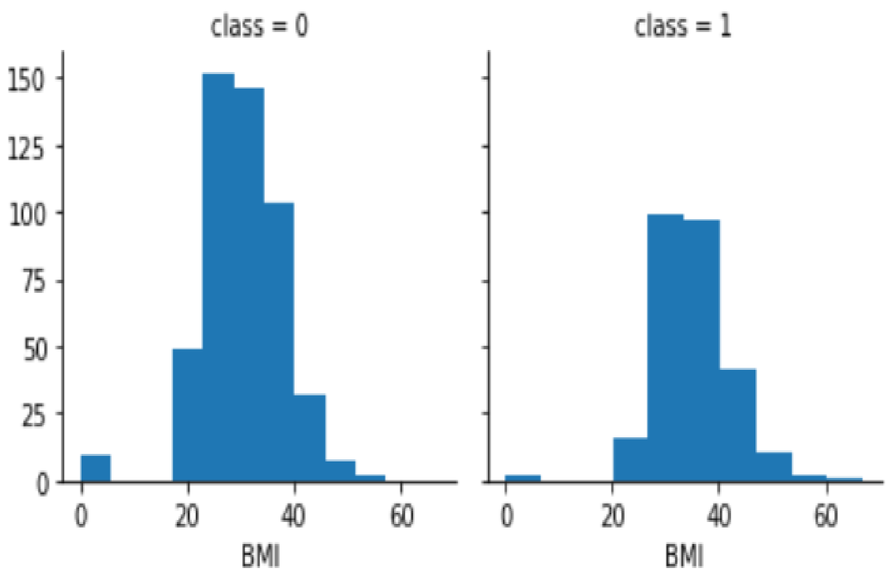

Figure 6. Distribution of people according to BMI

In the Pima Indian dataset, plasma and BMI are known to have a high correlation with diabetes. Figures 4,5, and 6 demonstrate this relationship.

3.3 Optimal model

The large number of parameters and complexity in deep learning models can make them susceptible to overfitting. To address this, techniques such as early stopping [11], dropout[12], and cross-validation techniques [13] can be used to improve the performance and generalization of the model. By combining these techniques, an optimal model that generalizes well to unseen data and has reduced overfitting can be obtained. Early stopping helps prevent overfitting during training, dropout regularizes the model and improves generalization, and cross-validation provides a more robust assessment of how well the model generalizes to unseen data. These techniques were adopted in the proposed model. When using early stopping and dropout together, it is important to find the proper balance between preventing overfitting and allowing the model to learn enough to achieve good performance on the training set. This involves experimenting with different hyperparameters such as the dropout rate, patience parameter, and learning rate to find the best combination for this problem. Early stopping, dropout, and cross-validation techniques are explained in the next section.

3.3.1 Early Stopping

Early stopping is a regularization technique commonly employed in machine learning algorithms, particularly in training neural networks. The goal of early stopping is to prevent overfitting and find the optimal point at which to stop the training process. During the training of a machine learning model, the performance on a validation set is monitored at regular intervals. The validation set consists of data that is separate from the training set and is not used for updating the model's parameters. The model's performance on the validation set is typically measured using a validation metric, such as accuracy or loss. Early stopping works by tracking the validation metric over the course of training. The training process continues until the validation metric reaches a certain threshold or starts to deteriorate. At that point, training is stopped, and the model's parameters are saved.

The rationale behind early stopping is that as the training progresses, the model initially improves its performance on both the training set and the validation set. However, after a certain point, the model might start to overfit the training set, causing the performance on the validation set to decline. Early stopping helps to find the sweet spot where the model performs well on both the training and validation sets, indicating good generalization capability. Stopping the training early before overfitting occurs, helps prevent the model from memorizing noise or specific patterns in the training data that do not generalize well to unseen data. When implementing early stopping, it is essential to balance the trade-off between stopping too early, which may result in an underfit model, and stopping too late, which may lead to overfitting. The optimal point for early stopping can be determined by monitoring the validation metric over multiple epochs and selecting the point where the metric starts to deteriorate or no longer improves significantly. Early stopping is a simple yet effective technique for regularization and can be applied to a wide range of machine learning algorithms, not just neural networks. It helps improve the generalization performance of models by finding the point where the model achieves the best trade-off between training and validation performance. The optimal patience parameter for early stopping in deep learning models depends on the specific problem, dataset, and model architecture. The patience parameter determines how many epochs to wait before stopping the training process if the validation loss has not improved. Setting the patience parameter too low can cause the training process to stop too early and result in underfitting, while setting it too high can cause the training process to continue for too long and result in overfitting. Generally, a patience parameter value between 5 to 10 is a good starting point, but the optimal value may need to be adjusted based on the problem.

3.3.2 Dropout

To address overfitting in deep learning models, another technique called dropout is used. It refers to a regularization technique commonly used in neural networks. It allows the model to focus on learning the essential features that lead to better generalization and improved performance on unseen data. Dropout is primarily applied during the training phase of the neural network. This technique randomly drops out or sets to zero some neurons in a layer during each training epoch. This approach reduces the model's dependence on a particular set of neurons, promotes the learning of more resilient features, and enables better generalization of the model to new data. During the testing or inference phase, dropout is typically turned off, and the full network is used to make predictions. The dropout technique has been found to be particularly effective in deep neural networks, where overfitting is a common challenge due to a large number of parameters. It allows neural networks to learn more robust and generalizable representations of the data. It is important to note that dropout is just one of several regularization techniques available in machine learning. Other techniques, such as L1 and L2 regularization, can be used in combination with dropout to further improve the model's performance and prevent overfitting. The optimal dropout rate depends on the specific neural network architecture and the complexity of the dataset being used. There is no one-size-fits-all answer for the optimal dropout rate, as it varies from one problem to another. Generally, the dropout rate can be set to a value between 0.2 and 0.5, with a default value of 0.5 often used as a starting point. This means that each neuron in the layer has a 50% chance of being dropped out during each training epoch. However, it is important to note that setting the dropout rate too high can cause the model to underfit the data while setting it too low can cause the model to overfit the data.

3.3.3 Cross validation

A popular technique used for cross-validation in machine learning is 10-fold cross-validation. This approach involves randomly dividing the dataset into 10 equal-sized subsets or folds. Compared to other forms of cross-validation, such as leave-one-out cross-validation or 5-fold cross-validation, 10-fold cross-validation generally provides a more precise estimate of the model's performance. This is because it uses a larger number of test sets, which helps to reduce the variance in the estimate of the model's performance. Another advantage of using 10-fold cross-validation is that it allows for a more efficient use of the data compared to other forms of cross-validation. Determining the optimal value for the number of folds in cross-validation depends on several factors, including the size of the dataset and the specific characteristics of the data. In general, 10-fold cross-validation is a commonly used choice as it provides a good balance between computational efficiency and reliable performance estimation. In this study, the performance was evaluated while varying the k value from 5 to 10. In the case of the Pima Indian dataset, when the k value was set to 10, it exhibited excellent performance. The performance variations with respect to the k value are shown in Table 3.

Table 3. Performance while varying the k value from 5 to 10

Number of folds	Accuracy
5	0.6953060011883541
6	0.6666666666666666
7	0.6783986655546289
8	0.6796875
9	0.692749658002736
10	0.699231032125769

3.4 Experiment

Classifying patients into one of three categories: normal, prediabetes, or diabetes, the MLP algorithm is trained on a dataset of patient data, including various features that are relevant to the prediction of diabetes. The MLP model learns the relationships between these features and the corresponding classification of the patient. Once the MLP model has been trained, it can then be used to predict the classification of new patients based on their data. The MLP takes the input features of a new patient and calculates the output probabilities for each of the three categories. The prediction of the patient's classification is based on the category with the highest probability. This is known as the "maximum a posteriori" (MAP) decision rule. In addition to the training phase, the MLP model requires setting the number of hidden layers and the number of neurons within each layer, which can impact the model's accuracy. In this study, two hidden layers were used and each layer has 12 and 8 nodes. The performance of the proposed model was evaluated based on data that was not used during the training process. Figure 7 shows the results of the experiment on the test data in the test data set. It shows the results of three different classifications of test data, including diabetes, normal, and pre-diabetes.

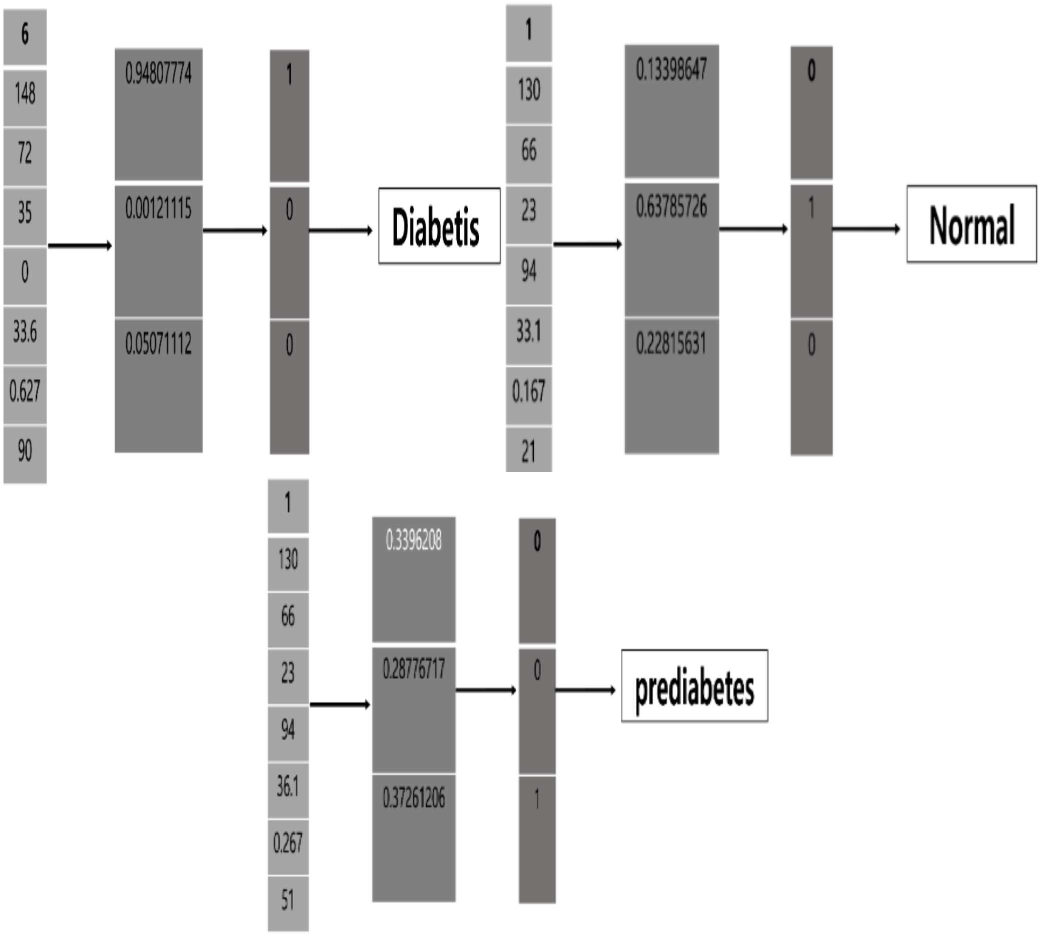

Figure 7. Evaluation of proposed method on unseen data.

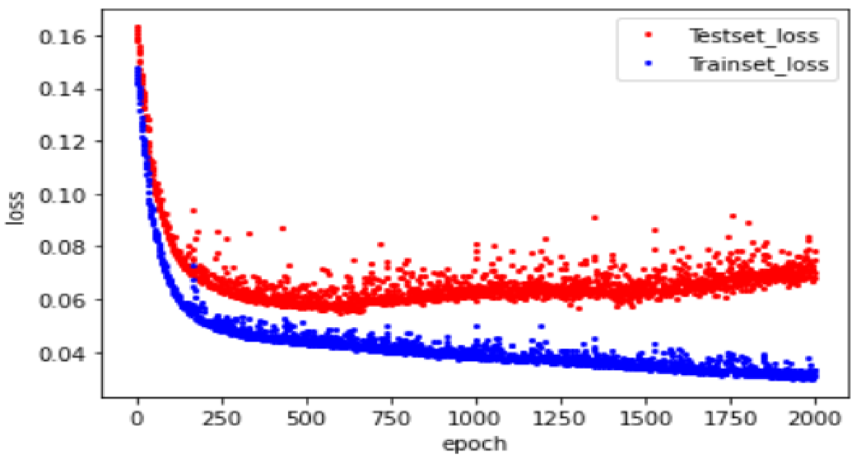

Figure 8. Changes in error value with respect to epochs in training and testing Data

3.5 Analysis of experimental result

Logistic regression and MLP are two popular machine learning algorithms that are widely used in the field of healthcare for diagnosis and prediction tasks. For the evaluation of the performance of a diabetes management system using logistic regression and MLP, recall[14], precision[15], and F1 score[16] were used. These metrics are commonly used in the evaluation of multi-class classification models and provide valuable insight into the performance of the system. Precision indicates the accuracy of positive predictions, while recall measures the coverage of positive examples. By using both recall and precision, a comprehensive understanding of the system's performance can be derived and areas for improvement can be identified. The F1 score is a metric used to evaluate the performance of a classification model, particularly when the classes are imbalanced. It combines the precision and recall of a model into a single score that represents the model's overall accuracy. In the case of the Pima Indian dataset, the occurrence of diabetes is generally the minority class compared to non-diabetic instances. This class imbalance can pose challenges for classification models as they may tend to favor the majority class and have lower performance in accurately predicting the minority class.

In this research, the precision, recall, and F1 score are 0.8607, 0.8813, and 0.871 respectively. This suggests that the classifier has a high accuracy for positive predictions (precision) and a high coverage of positive examples (recall), leading to a high F1 score. These results indicate that the classifier performed well in both correctly identifying positive cases and accurately identifying the proportion of positive cases in heavily skewed data.

4. Conclusion and future work

In this paper, a diabetes-predicting system based on machine learning techniques, specifically logistic regression and a multilayer perceptron (MLP) was proposed. The logistic regression algorithm was used to predict the probability of a patient having diabetes based on the patient's data. The MLP is then used to classify the patients into one of three categories: normal, prediabetes, or diabetes. These algorithms were trained using a dataset of patient data, including demographic information, laboratory results, and medical history.

The future research plans aim to develop ensemble models, predict and manage diabetes based on data from Koreans, and focus on ethical data management. Logistic regression and MLP can be integrated with other machine learning methods, such as decision trees, random forests, and gradient boosting, to improve performance and robustness. Ensemble methods can provide several benefits, including improved prediction accuracy, better generalization, and increased robustness to noise and outliers. They are commonly used in various machine learning tasks, such as classification, regression, and anomaly detection. However, ensemble methods can be computationally expensive and require more resources compared to individual models. They may also be more complex to interpret and may not always lead to significant improvements in performance, especially if the individual models are highly correlated or the data is biased. Overall, ensemble methods are a powerful technique in machine learning that can enhance predictive performance, but careful consideration should be given to the specific problem and the characteristics of the data to determine if and how to apply ensemble methods effectively.

Additionally, this dataset was collected from only one population, the Pima Indians, and may not generalize well to other populations with different ethnicities, lifestyles, or environmental factors. As such, care should be taken when applying models trained on this dataset to other populations. Another limitation of the dataset is that it only contains data for 768 patients and may not be representative of the entire population of individuals with diabetes. A larger, more diverse dataset would likely provide a more accurate picture of the relationship between risk factors and the development of diabetes. A diabetes dataset targeting Koreans is expected to be publicly released soon. This dataset will provide valuable information about diabetes in the Korean population, and it will be useful for developing predictive models and evaluating their performance.

Before using the Korean diabetes dataset, ethical considerations have to be done. Ethical considerations are important in any research or study, including in the field of diabetes management. Some ethical considerations that should be taken into account are as follows. First, it has to be ensured that participants in the study provide informed consent and understand the purpose of the study, potential risks and benefits, and their rights as participants. Second, privacy and confidentiality. Ensuring the privacy and confidentiality of participants' personal information and data is crucial. Adherence to data protection regulations should be observed and secure methods for data storage and transmission should be used. Third, data usage and ownership should be considered. The ownership and usage rights of the collected data will be clearly defined. Permission from participants to use their data for research purposes will be obtained and participants will be ensured that it is used responsibly and ethically. Fourth, bias and fairness will be considered. There should be no biases in the data collection, analysis, and reporting processes. All participants will be treated fairly and will be ensured that the study does not discriminate against any individual or group based on factors such as race, gender, or socioeconomic status. Fifth, potential harm will be considered. Any potential harm or risks to participants will be minimized. The participant will be ensured that the study procedures and interventions are safe and do not cause any unnecessary physical or psychological harm. Sixth, there has to be transparency and reporting. Clear and transparent reporting of the study methodology, results, and conclusions will be provided. Any conflicts of interest or funding sources that may have influenced the study will be clearly disclosed. Finally, an ethical review will be done. Ethical approval from relevant institutional review boards or ethics committees before conducting the study will be sought. Adherence to the ethical guidelines and standards set by the research community and regulatory bodies will be observed. By considering these ethical aspects, researchers can ensure that their study is conducted with integrity, respecting the rights and well-being of participants, and contributing to the advancement of knowledge in a responsible and ethical manner.

Acknowledgements

This work was supported by Youngsan University Research Fund 2023

References

1. Moon EJ, Jo YE, Park TC, Kim YK, Jung SH, Kim HJ, Kim DJ, Chung YS, Lee KW. Clinical characteristics and direct medical costs of type 2 diabetic patients. Korean Diabetes J. 2008. 32:358–365.
2. Kumar Dewangan, A., & Agrawal, P. (2015). Classification of diabetes mellitus using machine learning techniques. International Journal of Engineering and Applied Sciences, 2(5), 257905..
3. Agrawal, K., Bhargav, G., & Spandana, E. (2021). Diabetes Diagnosis Prediction Using Ensemble Approach. In Proceedings of the Fourth International Conference on Microelectronics, Computing and Communication Systems: MCCS 2019 (pp. 799-813). Springer Singapore.
4. Hosmer DW, Lemeshow S. Applied Logistic Regression. 2nd ed. New York, NY: John Wiley & Sons, Inc; 2000.
5. Accessable at https://analyticsindiamag.com/basics-of-ensemble-learning-in-classification-techniques-explained/
6. Accessable at https://scikit-learn.org/stable/modules/ensemble.html
7. Wen, Y., Tran, D., Ba, J.: Batchensemble: an alternative approach to efficient ensemble and lifelong learning. arXiv preprint arXiv:2002.06715 (2020)
8. Nelly David , Nathan S. Netanyahu Adaptive Consensus-Based Ensemble for Improved Deep Learning Inference Cost Lecture Notes in Computer Science book series (LNTCS,volume 12893) 2021
9. https://www.kaggle.com/datasets/uciml/pima-indians-diabetes-database
10. Mukhiya, S. K., & Ahmed, U. (2020). Hands-On Exploratory Data Analysis with Python: Perform EDA techniques to understand, summarize, and investigate your data. Packt Publishing Ltd.
11. Ji, Z., Li, J., & Telgarsky, M. (2021). Early-stopped neural networks are consistent. Advances in Neural Information Processing Systems, 34, 1805-1817.
12. Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., & Salakhutdinov, R. (2014). Dropout: a simple way to prevent neural networks from overfitting. The journal of machine learning research, 15(1), 1929-1958.
13. Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., & Salakhutdinov, R. (2014). Dropout: a simple way to prevent neural networks from overfitting. The journal of machine learning research, 15(1), 1929-1958.
14. Juba, B., & Le, H. S. (2019, July). Precision-recall versus accuracy and the role of large data sets. In Proceedings of the AAAI conference on artificial intelligence (Vol. 33, No. 01, pp. 4039-4048).
15. Michaud, E. J., Liu, Z., & Tegmark, M. (2023). Precision Machine Learning. Entropy, 25(1), 175.
16. Chicco, D., & Jurman, G. (2020). The advantages of the Matthews correlation coefficient (MCC) over F1 score and accuracy in binary classification evaluation. BMC genomics, 21, 1-13.
17. Safdar, N. M., Banja, J. D., & Meltzer, C. C. (2020). Ethical considerations in artificial intelligence. European journal of radiology, 122, 108768.