Diabetes Prediction Using Perceptron Algorithm

Mulugeta Lilay
The University of Adelaide
Adelaide, South Australia, Australia
mulugeta.lilay@student.adelaide.edu.au

Abstract

Diabetes is a fatal chronic disease that has been killing many people around the globe. Individuals with diabetes have many symptoms such as increased hunger, blurred vision, weight loss, repeated infections, etc. Early detection and management of the disease is important for effective treatment. And advanced machine learning and artificial intelligence algorithms can be used to develop models that can effectively identify diabetes. In this work, perceptron model is developed to predict diabetes. The Pima Indians diabetes dataset is downloaded and pre-processed to build and select a model. The developed model achieved 77% accuracy and 0.65 F1-score.

1 Introduction

Diabetes is a disease associated with an increase in blood glucose level in human body or deficiency in insulin which is the hormone that controls the level of glucose in our blood. This disease can severely affect different organs of human body, including the brain, nerves, heart, kidney, eyes and skin. It is a fatal chronic disease that has been killing many people. It is killing about 2-3 million patients every year (Mujumdar and Vaidehi 2019, p. 293). The number of people died due to diabetes in 2019 was approximately 4.2 million (Ganie et al. 2023, p. 2).

Individuals with diabetes have many symptoms, e.g., increased hunger, blurred vision, weight loss, numbness, repeated infections, etc. (Dharmarathne et al. 2024, p. 1). These symptoms are used to identify individuals who are in need of medical attention and further evaluation. Early detection and management of the disease is important for effective treatment. And this can be achieved by employing advanced methods that can effectively recognise the symptoms and signs of the disease.

Nowadays, the use of machine learning techniques in healthcare systems has improved disease diagnosis. This is due to the fact that machine learning techniques can produce improved diagnosis accuracy, treatment arrangement and patient care. Advanced machine learning and artificial intelligence algorithms can be used to develop models that can effectively identify diabetes (Ganie et al. 2023, p. 2). These algorithms analyse large-scale datasets to extract meaningful patterns and use them to make accurate predictions. With the availability of data, researchers have employed various machine learning algorithms such as Artificial

Neural Networks, Decision Trees, Random Forest, logistic regression, and so on to analyse the diabetes conditions and generate predictions (Dharmarathne et al. 2024, p. 2). Recently, neural networks have become the most valued tools by researchers. Perceptron is an artificial neural network invented by Frank Rosenblatt (Rosenblatt 1958) with simple structure and good generalization performance. It is a binary linear classifier that exhibits strict convergence for the problems it can solve.

In this work, perceptron algorithm is used to predict diabetes. We have employed Pima Indians diabetes dataset to train, validate and test a perceptron model. Evaluation metrics such as accuracy, precision, recall, F1-score and confusion matrix are used to evaluate the developed model. And the results are presented and discussed.

2 Methodology

In this section, the description of the dataset, data pre-processing, the perceptron algorithm, and the training strategies we used are discussed.

2.1 Dataset

Here, we have downloaded the Pima Indians diabetes dataset from Kaggle `https://www.kaggle.com/datasets/mathchi/diabetes-data-set?resource=download`(Annamoradnejad and Zoghi 2020). This dataset contains 8 feature columns and a binary outcome column labeled 0 or 1. The features include the number of pregnancies, glucose concentration, blood pressure, skin thickness, insulin level, Body Mass Index (BMI), age, and Diabetes pedigree function (DPF). And the dataset has 768 instances. The detail description of the dataset including, feature definition, measurement units and range of values for the features is given in table 1 (Ganie et al. 2023, p. 2). And feature distributions for all feature columns is given as histogram plots in figure 1 below.

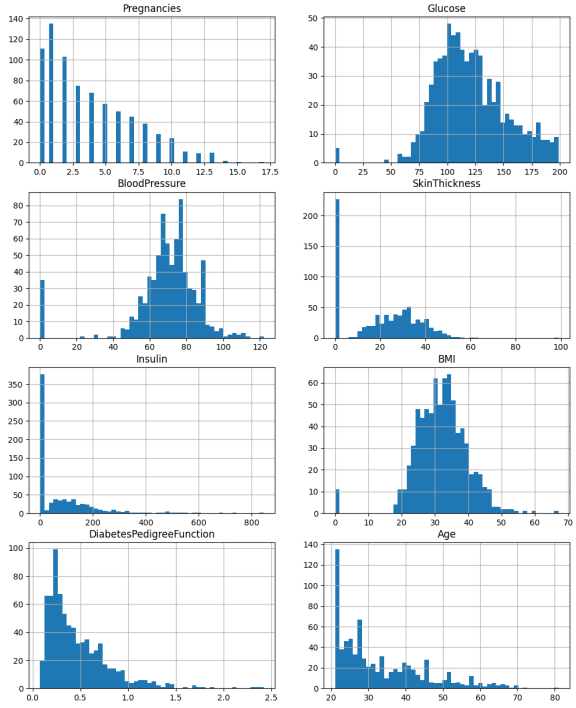

Figure 1: Feature distributions

2.2 Data Pre-processing

The dataset we downloaded has no missing values and the feature columns are all numeric data types. But we have identified that some features have outliers and non-normal distributions. Thus, in this subsection, we performed standard feature scaling as feature scaling is important to improve model performance, ensure feature values are in the same scale and minimize effects of outliers in a dataset. And we did standard scaling which makes the feature values to be centered around the mean with a

Table 1: Detailed description of dataset

Feature	Description	Value range
Pregnancy	refers to number of pregnancy times (numeric)	0-17
Glucose	refers to the participant's plasma glucose concentration measured in mg/dL	0-199
Blood pressure	Refers to blood pressure (mmHg)	0-122
Skin thickness	Participant's triceps skin fold thickness (mm)	0-99
Insulin	Participant's insulin level (2h - serum) measured in (mu U/mL)	0-846
Body Mass Index (BMI)	Participants body fat calculated using height and weight (kg/m2)	0-67
Diabetes pedigree function (DPF)	Diabetes likelihood based on family diabetes history (p-value)	0.07-2.24
Age	Age of participant (numeric)	21-81

unit standard deviation using the following formula(Nurdin et al. 2023, p. 995).

$$X_{new} = \frac{X_{old} - \mu}{\sigma}$$

where X_{new} is new feature value, X_{old} is old value of the feature, μ is mean of the feature column and σ is standard deviation of the column. Here, we have also split our dataset into training, validation and testing sets with a 70/20/10 splitting percentage. The dataset has 768 total records and the splitting step gives us 552, 139 and 77 instances in the training, validation and testing sets respectively.

2.3 Perceptron Algorithm

Binary classification is a type of classification problem in which input data is classified into one of two possible categories. In machine learning, binary classifiers are models trained to classify data into one of two classes. And the classes are represented as labels such as 0 or

1, true or false, or positive or negative. Binary classifiers are supervised type of machine learning models that they are trained with labelled data where the correct label for each training example is known. Then the trained models are used to predict labels or classes for new or unseen data.

The perceptron is the simplest type of artificial neural network and is a foundational building block of more complex models. It is a supervised linear classifier used to perform various binary classification tasks. The perceptron contains the following basic components:

1. Input layer consists one or more input neurons that accept input from the external world.

2. Weights are associated with each input neuron that represent strength of the connection to the output.

3. Bias bias term is added to the input layer to enable the perceptron modelling complex patterns in the input data.

4. Activation function produces the output taking the weighted sum of the inputs and the bias term.

5. Output a single binary value, 0 or 1 or -1 or +1, which indicates the class that the input data belongs.

6. Training algorithm iterative adjustment of the weights and bias parameters of the perceptron to minimize the prediction error for a given set of training examples. Supervised perceptron learning algorithm or backpropagation is used for training. The perceptron model is given in the following figure.

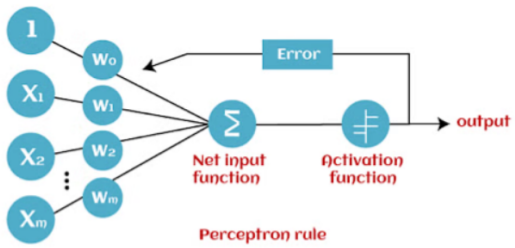

Figure 2: Perceptron model

As we can see from figure 2, the values $(X_1, X_2, ..., X_n)$ represent the input values, $(W_0, W_1, ..., W_m)$ represent corresponding weight parameters, and bias = 1. The net input function (Σ) multiplies each input value with its weight and then add these values to obtain the weighted sum as follows.

$$\Sigma = \sum_{i=1}^{m} X_i W_i + bias.$$

And the resulting weighted sum of the input values is then given to the activation function which produces an output (Y).

$$Y = f(\Sigma)$$

There are various activation functions such as step function, sigmoid and ReLU function. Training is then performed by updating the weights and bias using the following formula.

$$w_i^{\text{next step}} = w_i - \eta(y_i - \hat{y}_i)x_i$$

Where η is the learning rate, $x_i s$ are the input features and $y_i s$ are the output labels.

2.4 Training Strategies

In this work, we have developed a perceptron model which is a single layer neural network for diabetes prediction. The model has 8 input nodes, randomly initialized weights and sigmoid activation function. And to train the

model, we have used Stochastic Gradient Descent (SGD) optimizer as our optimization algorithm, binary cross entropy loss as a loss function, accuracy as a performance metrics. We have trained the model for a total of 200 epochs with early stopping criteria of 5 patience in validation losses and batch size of 32.

3 Experimental Analysis

During model building and selection, we have developed models with different hyperparameters setup using the training and validation data. We have employed the training dataset to train models with different learning rate values and evaluate them using the validation dataset to select the best model configuration. For model performance evaluation, we have used performance metrics such as accuracy, f1-score, precision, recall. And the formulas are given as follows (Mujumdar and Vaidehi 2019, p. 297).

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

$$Precision = \frac{TP}{TP + FP}$$

$$Recall = \frac{TP}{TP + FN}$$

$$F1 - Score = \frac{2 * precision * recall}{precision + recall}$$

For the hyperparameter selection, we have trained models with different learning rates (0.005, 0.001, 0.01, 0.1, 1.0) and evaluate and compare the trained models based on their mean validation accuracy to select the best learning rate for our final model. The model with learning rate = 0.01 outperforms the other models and we select the learning rate to be 0.01. The other hyperparameters are set to default for this work. After we have selected the optimal learning rate for our model, we have trained the final model with the optimized hyperparameters using the full dataset comprising the training and validation datasets together. Lastly, we have tested the final model using our test dataset we put aside at the beginning of the data splitting step. The learning curve for the final best model is given in figure 3 below.

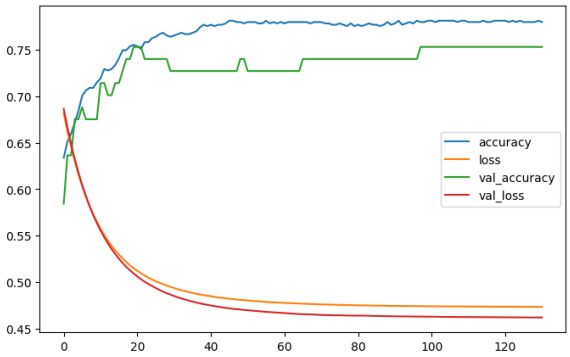

Figure 3: Selected model learning curve

As we can see from the curve, the model has a good convergence and stable learning. And the performance measures that the model achieved are given in the following table.

Table 2: Performance measures

Metrics	Score
accuracy	0.77
roc_auc	0.74
precision	0.63
recall	0.68
f1-measure	0.65

The model achieves 77% accuracy and 0.65 f1-measure. In this case, f1-measure can be a good evaluation metric as it balances the trade-off between true positive rate and precision. In addition, we have used confusion matrix to evaluate the model and it is given in figure 4 below. The confusion matrix shows that out of 52 instances with label 0, 42 are correctly predicted

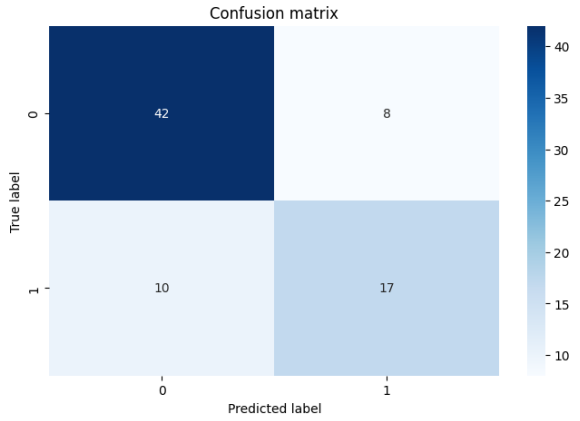

Figure 4: Confusion matrix

and out of 25 instances with label 1, only 17 are correctly predicted. This shows that the model predicts data with label 0 better than predicting data with label 1.

4 Conclusion and Future Work

The aim of this work was to implement diabetes prediction using perceptron algorithm. We have used Pima Indians diabetes dataset to develop perceptron model. The dataset is split into training, validation and testing sets. Hyperparameter optimization is performed to select the best model configuration and then the selected model is tested on the testing dataset. The selected model achieves 77% accuracy and 0.65 f1-score. The obtained results show that the developed model has fair performance to predict diabetes from input data. But further works can be done to achieve more promising prediction results. Using larger size dataset, data balancing and developing models with high level of flexibility can be recommended directions for future work.

5 Code

The code for the paper and the dataset used are available at `https://github.com/MYikuno/Fundamentals-of-DL.git`

References

Annamoradnejad, Issa and Gohar Zoghi (2020). "Colbert: Using bert sentence embedding for humor detection". In: *arXiv preprint arXiv:2004.12765*.

Dharmarathne, Gangani et al. (2024). "A novel machine learning approach for diagnosing diabetes with a self-explainable interface". In: *Healthcare Analytics* 5, p. 100301.

Ganie, Shahid Mohammad et al. (2023). "An ensemble learning approach for diabetes prediction using boosting techniques". In: *Frontiers in Genetics* 14, p. 1252159.

Mujumdar, Aishwarya and Vb Vaidehi (2019). "Diabetes prediction using machine learning algorithms". In: *Procedia Computer Science* 165, pp. 292–299.

Nurdin, Averina et al. (2023). "Using Machine Learning for the Prediction of Diabetes with Emphasis on Blood Content". In: *Procedia Computer Science* 227, pp. 990–1001.