Diabetes prediction using Perceptron

Shivangi Patel
The University of Adelaide
SA 5005 Australia

Abstract

Diabetes is a chronic disease with global prevalence, resulting in more than 2 million deaths annually[1]. Type-2 diabetes, which comprises 85-90% of all diabetes cases, often goes undiagnosed for several years from onset[2]. The aim of this study was to predict diagnosis of Type-2 diabetes using a supervised machine learning approach. PIMA Indian diabetes dataset was used for the study. Two models - a Basic Perceptron and a Multi Layer Perceptron (MLP) were trained and validated using 5-fold cross validation technique. The mean classification accuracy of MLP was found to be higher than Basic perceptron on a subset of data; and therefore it was further evaluated on unseen 'test' data. The trained MLP model was able to predict diagnosis of diabetes with 76% accuracy.

1. Introduction

Type-2 diabetes is a chronic disease condition where the body either fails to produce enough insulin; or progressively becomes resistant to the effects of insulin. When insulin resistance commences, as a response to manage blood glucose levels, the insulin producing cells (β cells) in pancreas wear themselves out due to overproduction of insulin. The condition worsens with loss of up to 70% of β cells; and may result in severe long term implications if not diagnosed timely [2]. Type-2 diabetes is largely influenced by lifestyle choices, genetic risk factors and ethnicity background. It usually develops in adults over 45 years; but is also increasingly found in younger age groups. Glucose and insulin levels in blood are also good indicators of disease onset. The Pima Indians (tribe residing in deserts of Arizona, USA) have been under epidemiological studies since 1965 by the National Institute of Diabetes and Digestive and Kidney Diseases due to exceptionally high prevalence of diabetes in the population [4,5]. With several risk factors associated with type-2 diabetes identified from decades of research, it serves as a good candidate for predictive diagnosis using supervised machine learning techniques.

2. Related Research

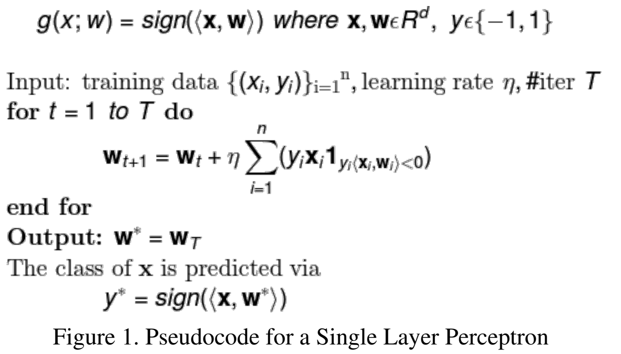
In the past three decades, several studies aiming to predict diagnosis in the Pima Indian dataset, have been reported with promising results. Most of the published reports employed statistical modelling or traditional machine learning techniques, while few researchers have focused on deep learning. The findings of a few selective studies have been summarized in this section. Harleen K. et al[7] imputed missing data using K-Nearest Neighbours predictions, followed by a Boruta wrapper for feature selection. K-Nearest Neighbours is an instance based non-parametric method, which when employed for missing value imputation finds nearby samples in the training set and averages them to fill the value [8]. Boruta is designed around a Random Forest classification algorithm; and iteratively removes less relevant features based on certain statistical tests [6]. They reported 88% classification accuracy using Support Vector Machine (linear kernel). Originally developed by Vapnik and Chevronekis in 1974, SVM with a linear 'kernel' function is a non-probabilistic binary classifier that strives to obtain an optimal hyperplane between linearly separable classes. An integration method between K-means clustering algorithm and SVM was proposed by Ahmed H. et al [9]. K-means is an unsupervised algorithm that groups similar data points in a dataset to form k clusters [10]. The clustering output of K-means unsupervised learning was utilized as input for the SVM classifier; enhancing the classification accuracy to 99%. Asma A. et al [11] performed numerical discretization of features, followed by training a J48 Decision Tree classifier using Weka software [12]; obtaining an accuracy of 78%. Finally, a multi layered neural network architecture trained using Levenberg-Marquardt (LM) algorithm was reported by Hasan et al [13]; claiming an accuracy of 82%. LM is an optimization algorithm solving the problem of least squares curve fitting [14,15].

3. Methodology

Pima Indian diabetes dataset was downloaded from Kaggle. The dataset comprises 8 diagnostic measurements collected from 768 female subjects aged 21 years and above. It is a preprocessed dataset where several missing values were imputed as zero. Biologically, a zero value for 'Blood pressure', 'Glucose' and 'BMI' are not possible for living subjects; and therefore the corresponding rows were deleted, resulting in 724 rows. The 'Outcome' variable comprised two classes labeled as 0 and 1 for 'non-diabetic' and 'diabetic' subjects respectively. Class 0 was replaced with -1 so as to be able to use $sign$ function during prediction. After splitting the data into train and test sets in a 70: 30 ratio, the features were scaled from 0 to 1 using fit-transform method of sklearn StandardScaler function[18]. Two neural network models were investigated - a basic or Single Layer Perceptron (SLP) with a linear activation function; and a Multi Layer Perceptron (MLP) with non linear activations.

3.1. Single Layer Perceptron

A simplest simulation of a neural network comprises a single neuron with linear activation function as originally proposed by Rosenblatt F. [16]. Briefly, the perceptron takes inputs as a vector of features. Each feature is associated with a 'weight' value and thus there are as many numbers of weights as there are features. A single bias term is usually incorporated into the weight vector or it can be added as a separate variable. Typically weights and bias are initialized to zero in a SLP; however we used small positive random numbers initialisation for weights; while bias was initialised to zero. The weighted inputs are summed and added to the bias, and this sum determines the output of neurons. The output is a $sign$ function; with values > 0 classified as $+1$, else -1. Weights and bias are learned by Stochastic Gradient Descent optimization. The pseudocode for basic perceptron constructed in this study is represented in Figure 1.



```
$g(x; w) = sign(\langle \mathbf{x}, \mathbf{w} \rangle)$ where $\mathbf{x}, \mathbf{w} \epsilon R^d$, $y \epsilon \{-1, 1\}$

Input: training data $\{(x_i, y_i)\}_{i=1}^n$, learning rate $\eta$, #iter $T$
for $t = 1$ to $T$ do
    $\mathbf{w}_{t+1} = \mathbf{w}_t + \eta \sum_{i=1}^{n} (y_i \mathbf{x}_i \mathbf{1}_{y_i \langle \mathbf{x}_i, \mathbf{w}_i \rangle < 0})$
end for
Output: $\mathbf{w}^* = \mathbf{w}_T$
The class of $\mathbf{x}$ is predicted via
    $y^* = sign(\langle \mathbf{x}, \mathbf{w}^* \rangle)$
```

Figure 1. Pseudocode for a Single Layer Perceptron

3.2. Multi Layer Perceptron

MLP is a feedforward and backpropagation neural network, with one or more hidden layers. Each hidden layer typically consists of multiple neurons that transform input from the previous layer using a non linear activation function. The 2-layer network architecture or a one hidden layer MLP used in this study maps a function $f : R^D \rightarrow R^L$, where D is the size of input vector x and L is the size of the output vector $f(x)$, such that in matrix notation:

$$f(x) = G(b^2 + W^2(s(b^1 + W^1x)))$$

with bias vectors b^1, b^2; weight matrices W^1, W^2 and activation functions G and s. Weight matrices are initialised to small random positive numbers; while bias vectors to zeroes. The architecture is represented in Figure 2. The vector $h(x) = s(b^1 + W^1x)$ constitutes the hidden layer. $W^1 \epsilon\ R^{D \times D_h}$ is the weight matrix connecting the input vector to the hidden layer. Each column W^1_{i} represents the weights from the input units to the $i - th$ hidden unit. The activation function s is the Rectified Linear Unit ($ReLU$)[17] where $ReLU(z) = max(0, z)$, and G is logistic sigmoid such that $sigmoid(z) = 1/(1 + e^{-z})$. To train the MLP, the parameters set $\theta = \{W^2, b^2, W^1, b^1\}$ is learned by Batch Gradient descent optimization. The gradients of the loss function L with respect to the parameters set $\frac{\partial L}{\partial \theta}$ is achieved through the backpropagation algorithm. Wherever feasible, a vectorized implementation to process all training examples is performed to ensure an efficient computation.

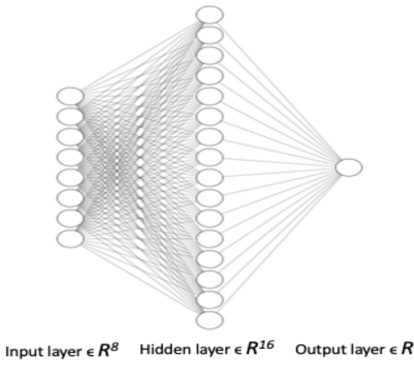

Input layer ϵ R^8
Hidden layer ϵ R^{16}
Output layer ϵ R^1

Figure 2. Architecture of Multi Layer Perceptron

4. Results and Discussion

The dataset comprised 724 rows after removing instances with zero values of 'Blood Pressure', 'Glucose' and 'BMI'. An imbalance in the classes was observed, with the ratio for diabetic to non diabetic subjects being approximately 35: 65. A heatmap of all features revealed 'Glucose' and 'Skin thickness' correlating highest and lowest respectively with the 'Outcome' variable. After splitting the dataset into train and test sets (70: 30), both the models - Single Layer Perceptron and Multi layer Perceptron were evaluated by 5-fold cross validation on the train set. Typical values of k for k-fold cross validation are 5 or 10; with higher values of k providing higher confidence in estimating the error. However, a very large k also means there are less number of instances in each fold and the folds may not be representative of the dataset. Choice of k should thus be based upon a careful consideration depending upon the size of the dataset. Figure 3 shows that with $k = 5$, the distribution of classes in each fold is uniform and representative of the entire dataset.

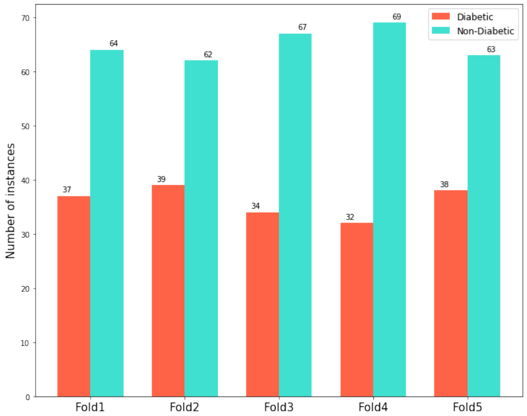


Figure 3. Distribution of classes in each fold

The SLP model gave a mean accuracy score of 70% with a standard deviation of 0.04, by 5-fold cross validation on the train set. This is a small improvement over a naive classifier, classifying all instances to the most frequent class $i.e.$ non-diabetic; which would give an accuracy of 65%. Since the SLP model computes only a linear activation, the low classification accuracy is indicative of the fact that the dataset is not linearly separable. We therefore investigated an MLP model consisting of one hidden layer. With $ReLU$ activation function for nodes in the hidden layer and a $sigmoid$ activation for the output layer, and a significant increase in the overall complexity of the neural network architecture; the MLP model gave a cross validation score of 76% and 0.02 standard deviation.The loss-curves of train and validation folds were monitored. It was observed that the training loss did not drastically decrease after 1000 iterations. The validation loss flattened at about 700 iterations and remained higher than the train loss.

Although the performance of MLP model is better than SLP; there are indications of both underfitting and overfitting the data. In other words,the model is unable to learn meaningful patterns in the train data while also somehow manages to capture the noise. Upon evaluation on test data, the trained MLP model was able to achieve a classification accuracy of 78%. The training of MLP may be further enhanced by tuning of hyperparameters such as the learning rate, number of hidden layers and nodes, activation functions in hidden layers and choice of optimization algorithm.

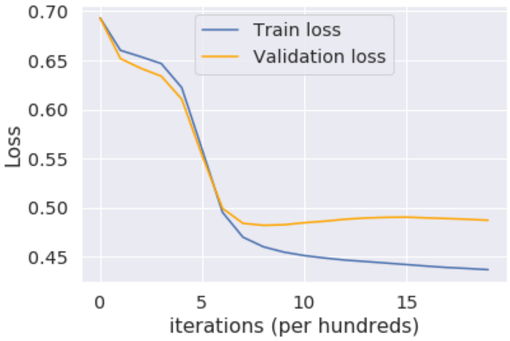


Figure 4. Loss curves for train and validation sets in a representative cross validation fold

5. Conclusion

A Single Layer Perceptron and a Multi Layer Perceptron with one hidden layer were investigated for classification of diabetes outcome in the Pima Indian diabetes dataset. Using only a linear activation function with SLP gave a poor cross validation accuracy; indicating the data may not be linearly separable. This was further validated by achieving a higher an enhanced performance with a Multi Layer Perceptron employing non-linear activation functions such as $ReLU$ for hidden layer and $sigmoid$ for the output layer. A further improvement in MLP model may be achieved with appropriate tuning of hyperparameters.

6. Github code

```
https://github.com/CrazyDaffodils/
Deep_learning_fundamentals/tree/master/
code
```

References

[1] World Health Organization: Diabetes, https://www.who.int/news-room/fact-sheets/detail/diabetes

[2] Diabetes Australia https://www.diabetesaustralia.com.au

[3] Kaggle: Pima Indian Diabetes dataset https://www.kaggle.com/uciml/pima-indians-diabetes-database

[4] Bennett PH, Burch TA, Miller M. *Diabetes mellitus in American (Pima) Indians.*. Lancet. 1971;2(7716):125-128. doi:10.1016/s0140-6736(71)92303-8

[5] Knowler WC, Bennett PH, Hamman RF, Miller M *Diabetes incidence and prevalence in Pima Indians: a 19-fold greater incidence than in Rochester, Minnesota*. Am J Epidemiol. 1978;108(6):497-505. doi:10.1093/oxfordjournals.aje.a112648

[6] Miron B. Kursa, Witold R. Rudnicki *Feature Selection with the Boruta Package*. Journal of Statistical Software, Articles. 2010;36(11):1-13. doi:110.18637/jss.v036.i11

[7] Harleen K., Vinita K. *Predictive modelling and analytics for diabetes using a machine learning approach.*. Applied Computing and Informatics. 2020;2634-1964

[8] Olga T., Michael C., Gavin S., Pat B., Trevor H., Robert T., David B., Russ B. *Missing value estimation methods for DNA microarrays*. Bioinformatics, 2001;17(6):520-525. https://doi.org/10.1093/bioinformatics/17.6.520

[9] Hamza O., Ahmed Moetque, Hani *Diabetes Disease Diagnosis Method based on Feature Extraction using K-SVM*. International Journal of Advanced Computer Science and Applications, 2017;8(10):14569

[10] James B. *Some methods for classification and analysis of multivariate observations*. Mathematics, 1967

[11] Asma A. *Decision tree discovery for the diagnosis of type II diabetes*. 2011 International Conference on Innovations in Information Technology, Abu Dhabi, 2011,303-307, doi: 10.1109/INNOVATIONS.2011.5893838

[12] Weka: The workbench for machine learning, https://www.cs.waikato.ac.nz/ml/weka/

[13] Hasan T., Nejat Y., Feyzullah T. *A comparative study on diabetes disease diagnosis using neural networks*. Expert Systems with Applications, 2009;36(4) 8610-8615, https://doi.org/10.1016/j.eswa.2008.10.032

[14] Kenneth L. *A method for the solution of certain non-linear problems in least squares*. Quarterly of Applied Mathematics, 1944;2(2) 164-168

[15] Donald W. *An Algorithm for Least-Squares Estimation of Nonlinear Parameters*. SIAM Journal on Applied Mathematics, 1962;11(2) 431-441

[16] Rosenblatt F. *The Perceptron: A probilistic model for information storge and organization in the brain.*. Psychological Review, 1958;65(6), 386-408

[17] Vinod N., Geoffrey E. *Rectified Linear Units Improve Restricted Boltzmann Machines*. http://www.cs.toronto.edu/ fritz/absps/reluICML.pdf

[18] Pedregosa, F., Varoquaux, G., Gramfort, A.,Michel, V., Thirion, B.,Grisel, O.,Blondel, M., Prettenhofer, P.,Weiss, R.,Dubourg, V.,Vanderplas, J., Passos, A.,Cournapeau, D.,Brucher, M.,Perrot, M.,Duchesnay, E *Scikit-learn: Machine Learning in Python*. Journal of Machine Learning Research, 2011;12, 2825-2830