Application of Deep Neural Networks in Diabetes Prediction Development

Ashmita Nigam | nigam.as@husky.neu.edu

***Abstract* —Machine learning and Deep Learning are popular technologies which are innovating the world, predicting outcomes and plays an important role in making good decisions. In a sector like healthcare, it is interesting to note how deep learning and machine learning can be beneficial in diagnosis and treatment of diseases like Diabetes.**

Diabetes is one of the most popular disease affecting the patients due to lack of insulin content in the body. Through this project, we aim to employ different machine learning techniques and deep learning techniques to identify important diabetes biomarkers and based on other data identify whether person has diabetes or not and compare the performances of these models. We also aim to develop an efficient model which can make good predictions.

Keywords

Deep Learning, Deep Neural Networks, AUC, Hyperparameter tuning, Diabetes prediction, Biomarkers,

Gradient Boosting Algorithm (GBM),

TensorFlow, Keras

Prediction,

Classification

1. Introduction

Machine Learning and Deep Learning are the subsets of artificial intelligence. These are the algorithms which can parse the data, learn from the data and then apply their learning to make informed decisions. Machine learning and deep learning techniques are being used for image recognition, classification, image classification, regression problems and used in varied business domains. Here we have taken a healthcare related diabetes dataset and tried to identify different biomarkers in diabetes and predicting whether person will have diabetes or not depending upon the other features present in the dataset. To achieve this, we have used different machine learning techniques such as Random Classifier, Gradient Boosting algorithm, K-Nearest Neighbors classifier and developed artificial neural network models to predict the outcome. We have used different evaluation metrics to evaluate different models such as Accuracy and AUC-ROC. We have plotted various performance related curves to understand the performance of these models better. Deep Learning Models of varied layers and different values of hyperparameters are used to improvise the model and achieve good accuracy score and minimize the losses as much as we can. In order to implement deep learning models, TensorFlow and Keras libraries are used.

2. Dataset

Diabetes Mellitus is a disease caused by excess of glucose in blood content and insulin which is a hormone produced by the pancreas in the human body are not enough. Due to unhealthy lifestyle, diabetes mellitus has become popular these days. It is important for doctors or hospitals to predict based on the current data and different biomarkers whether person will have diabetes or not and accordingly they can plan treatment and further diagnosis plan. This dataset has been formed by taking data of 768 people.

Dataset consists of 768 records and has 9 attributes. This dataset has following attributes:

- *BMI:* BMI of a patient
- *Insulin*: insulin measure in a patient's body.
- *Blood Glucose*: Glucose content in blood
- *Age*: Age of a person
- *Diabetes Pedigree*:
- *Skin Thickness*: Thickness level of skin in a patient
- *Blood Pressure*: Blood pressure measured
- *Pregnancies*:
- *Outcome*: This is the target

This is a classification dataset. We will be predicting the outcome of the variable ***Outcome*** which has values as 1 or 0.

Dataset consisted of various numeric, binary categorical columns.

Source:

https://www.kaggle.com/uciml/pima-indians-diabetes-database

3. Data Processing

In statistics, Exploratory Data Analysis is an approach to analyzing the data sets to summarize their main characteristics, often in visual graphical format. This also helps in analyzing the data and transforming the data in such a way that machine learning model can be built and gives better predictions. Exploratory data analysis involves data cleaning, transforming the data and plotting graphs to find relationship between variables, find correlations between variables. During our data cleaning tasks, we found following things:

There were no missing values present in the dataset.

We have normalized the data and performed feature scaling.

Scaling of features was important as it helps to normalize the data in a particular range and also enhance in speeding up calculations in algorithm.

4. Code and Documentation

GitHub Link: The complete code with documentation can be found on the below link:

https://github.com/ashmitan/Adv-in-Data-Science-Final-Project

Code can be divided into following parts:

1. Exploratory Data Analysis code
2. Machine Learning Techniques
   a. Random Forests Classifier
   b. K-Nearest Neighbors
   c. Gradient Boosting Algorithm
3. Deep Learning Models
   a. Model with 1 hidden layer
   b. Model with 5 hidden layers
   c. Model with multi layers

Random Forest Classifier:

We used Random Classifier method which is a popular machine learning technique. Random forest classifier creates a set of decision trees from randomly selected subset of training set. It then aggregates the votes from different decision trees to decide the final class of the test object.

Hyperparameters for Random Forests Classifier: Total Number of trees to be generated and decision tree related parameters like minimum split, split criteria etc.

K-Nearest Neighbor Classifier:

K-Nearest neighbor classifier is a popular algorithm useful for classification problems.

K-nearest neighbor algorithm predicts the class of the data point as per the majority of the votes obtained from the neighboring points and calculates distance such as Euclidean distance, hamming distance, cosine distance etc. Based on the votes, label is assigned to the new data point which needs to be predicted.

Gradient Boosting Algorithm:

Gradient boosting is a machine learning technique for regression and classification problems, which produces a prediction model in the form of an ensemble of weak prediction models, typically decision trees.

It builds the model in a stage-wise fashion like other boosting methods do, and it generalizes them by allowing optimization of an arbitrary differentiable loss function.

Deep Neural Network with 1 hidden layer:

Developed a neural network model. Here, we have defined a model having 3 dense layers. First is the input layer wherein we have defined an input shape which is 8 and defined activation function. I have used RELU activation function and as 12 neurons.

Deep Neural Network with 5 hidden layers:

Developed a neural network model. Here, we have defined a model having 3 hidden dense layers. First is the input layer wherein we have defined an input shape which is 8 and defined activation function. I have used RELU activation function. Used sigmoid activation function in the output layer. Ran the model for different number of epochs and different learning rates.

Deep Neural Network with more hidden layers:

Developed a neural network model. Here, we have defined a model having 7 hidden dense layers. First is the input layer wherein we have defined an input shape which is 8 and defined activation function. In the intermediate layers, we have used RELU activation function. Used sigmoid activation function in the output layer. Ran the model for different number of epochs and different learning rates.

5. Results

The dataset was selected, and we used Keras and TensorFlow library to develop neural network models. Target variable is ‘Outcome’.

We used Random Forests Classifier and achieved an accuracy of **72.3%.** When we tried K-nearest neighbors classifier technique then we achieved an accuracy of **75.3%** . Boosting algorithms usually help in improving the performance of the model. Gradient Boosting Algorithm achieved best accuracy amongst other machine learning technique of **78.3%.**

Several neural network models of varying depth were developed.

Firstly, developed a neural network model with 1 hidden layer along with input and output layer. We tried running the model with varied number of epochs starting from 200, 400 , 600 and so on. This model helped us achieve the best accuracy of **77.5%.**

Second model was developed with 2 number of hidden layers which helped in fetching accuracy of **73.2%** and the AUC score was **0.774**. This model was also used for different number of epochs.

Third model was developed with a greater number of hidden layers and varied number of neurons, modified the learning rates and optimizers and achieved best accuracy of **76.2%** and AUC-ROC score is **0.769**

If we used regularization techniques, GridSearchCV and tune other hyperparameters, we can get better results.

6. Conclusion

We found out various biomarkers in the diabetes dataset and we have used various techniques to improve the predictability of the model.

We have tried depicting how neural network models can be useful in healthcare, especially in diabetes dataset. We have tried comparing the performance of neural network models over traditional machine learning models.

We can develop different neural network models with varying number of layers and setting hyperparameters of the neural network model such as epochs, steps_per_epoch, learning rates etc.

After experimenting with different models, realized that model can learn the data only till certain number of epochs and then accuracy and other evaluation metric scores are constant. In order to improve the model, it is important to tune the hyperparameters and use effective optimizers such as SGD, Adam or RMSProp. Learning rates play an important role in performance of the model and different rates can bring a change in accuracy of the model.

7. Future Scope

In the future, if we test these models of varying number of layers, learning rates and use L1 and L2 regularization techniques can help in improving the performance of the neural network models.

Similar models can be built and run on other healthcare related dataset.

8. Acknowledgment

We are thankful to our Prof. Nik Bear Brown, Assistant Professor, and Prabhu Subramanian, Teaching Assistant, Northeastern University, Boston and, MA for their valuable guidance, encouragement, and co-operation during the implementation of this project.

9. References

[1] https://www.kaggle.com/uciml/pima-indians-diabetes-database

[2] https://www.geeksforgeeks.org/python-how-and-where-to-apply-feature-scaling/

[3]https://medium.com/machine-learning-101/chapter-5-random-forest-classifier-56dc7425c3e1

[4]https://towardsdatascience.com/machine-learning-basics-with-the-k-nearest-neighbors-algorithm-6a6e71d01761

[5]https://www.jeremyjordan.me/nn-learning-rate/

[6] https://machinelearningmastery.com/rectified-linear-activation-function-for-deep-learning-neural-networks/

[7]https://towardsdatascience.com/understanding-learning-rates-and-how-it-improves-performance-in-deep-learning-d0d4059c1c10

[8]https://www.kaggle.com/adhishthite/pima-dataset-prediction-model-with-keras-80

[9] https://keras.io/optimizers/