DIAGNOSING PARKINSON'S DISEASE USING RANDOM FOREST TECHNIQUE IN MACHINE LEARNING

Arpit Banerji, Lalit Sharma

Galgotias University, Greater Noida

Arpitbanerji9161@gmail.com

Abstract*: *Parkinson's Disease (PD) is a degenerative neurological disorder marked by decreased dopamine levels in the brain. There is no such laboratory test to diagnose PD, it is often very difficult to diagnose it, mainly in the early stages when effects are not yet severe. Monitoring the progression of the disease over time requires repeated clinic visits by the patient. In this paper, we have used the power of Machine Learning and Deep Neural Network to build a model for the detection of the disease and also ensemble techniques to improve the prediction accuracy. Also, model is validated using different metrics that are present in like confusion metric and accuracy score.

Keywords: PD, Deep Neural Networks, Machine Learning.

I. INTRODUCTION

Parkinson's disease (PD), a neurodegenerative disorder affects the neurons in the human brain. These cells generate a chemical in the human brain called dopamine, which is used to control all the movements in a human body. With the decrease of dopamine in the human body, it causes some uncontrollable movements in the body [1].

Parkinson's disease is affecting a significant percentage of human population around the world. It stays inside a human body and resides there for a long time and this is the reason why it is difficult to identify in an early stage. There are various symptoms of the disease which occur generally with varying stages of PD. Some of the normal symptoms of PD include mild to heavy tremors in a human body, speech disorders, unconditional facial expressions, etc. All these symptoms combined together may up to some extent concludes the presence of PD [1].

These symptoms can be tracked using various sensors for diagnosing voice recordings automatically. It is the most easiest and feasible way to detect the disease in the staring phase.

With the growing advancements in the field of health-care Machine Learning and Ai is playing a vital role in this domain. Hence we are trying to use Machine learning Techniques and Deep Neural Networks to detect Parkinson's Disease in an. Early-stage. The data-set we are using for the detection of the Parkinson's disease consists of vocal recordings with different sensors which can detect Speech disorders.[2]

With the help of Ensemble Machine Learning Techniques and Deep Neural Networks, we are trying to classify people between those who are suffering from PD and those who are suffering from Parkinson's Disease.

II. DESCRIPTION OF THE PROPOSED TECHNIQUES

In this paper we are using a classical Machine learning algorithm - Decision Tree Classifier, Machine Learning Ensemble Techniques - Random Forest Classifier and finally Deep Neural Networks to diagnose PD in an early stage.

We collectively compare the scores of the models to come up with the best scoring model. We use different scoring metrics such as F1- score, and accuracy to evaluate the model.

III. EMPIRICAL STUDIES

A. Dataset

Link for Dataset:
https://archive.ics.uci.edu/ml/datasets/parkinsons

The Parkinson's Disease dataset used in the work consists of a range of vocal measurements from 31 people, 23 with Parkinson's disease (PD). Each column in the table is a particular voice measure, and each row corresponds to one of 195 voice recordings from these individuals. The main aim of the data is to discriminate healthy people from those with PD, according to the "status" column which is set to 0 for healthy and 1 for PD. The data is in ASCII CSV format. The rows of the CSV file contain an instance corresponding to one voice recording. There are around six recordings per patient, the name of the patient is identified in the first column. [3]

B. Exploratory Data Analysis

The data is loaded into the data-frame and proper EDA is done on the dataset to get the insights of the data. We saw the correlation of the data-points with each other. We plot the data visually so that we can get further insights into the dataset. The histograms give a proper data distribution of the data. As we can see that there is an imbalance in the data-set but we are giving the model variance with making the class balance so that it can perform well with the given dataset.[4]

We have used box-plot to check the presence of outliers and we came to the conclusion that there were very few outliers so we didn't treat those outliers.

We have checked the correlation between the data and the target variable to check the correlation between the status and other independent variables and we found that all the variables contributed some correlation to the status, some positive and some negative so we are dropping no columns.

C. Model Development

We have split the data in 70:30 using train_test_split from the sklearn library. Where we are using 70 percent of the data in training the model and 30 percent of the data in testing the model.

- The first model we have implemented is a Decision Tree Classifier[5]. We have set the hyper-parameters to default and we have using the criterion of 'entropy'.
  The model gives an accuracy of 89.8 percent.
  F1-score of 92 percent.
  Confusion matrix is as follows: Figure 1

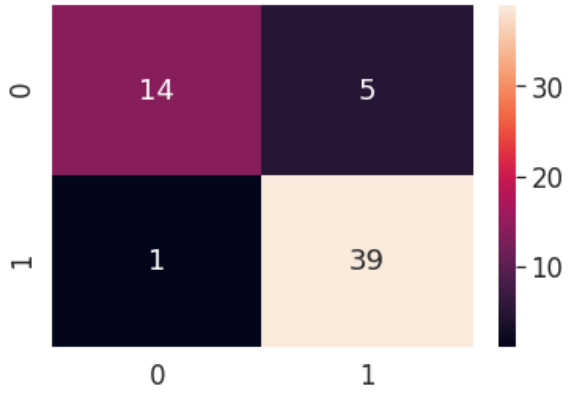

Figure 1

- The second model we used is a decision tree model with a max depth of 5 and max leaf nodes of 2.
  It gave an accuracy of 81 percent
  F1 -score of 86 percent and confusion matrix is as follows:

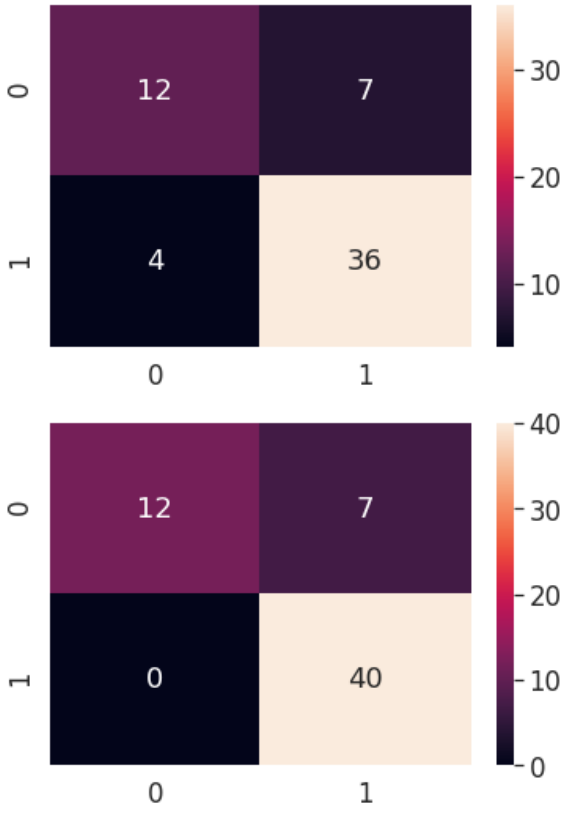

Figure 2

- The third model we used is a decision tree [5] with max depth and leaf nodes 5.
  It gave an accuracy of 86 percent.

F1 score of 90 percent and confusion matrix as follows:

Figure 3

As you can see from the above models, the ideal no of leaf nodes is between 10 to 15. The ideal no of leaf_nodes and depth of the tree can be found out by running these model iteratively by tuning the hyperparameters. In the above case, I have showcased how accuracy values change for few cases. It would be a right call to use Grid search or ROC curve (which you will be learning later as a part of this program) to find out the optimal hyperparameters.
Now we'll use Ensemble techniques i.e. Random Forest classifier.

- Random forest [6] gave an accuracy of 89 percent.
  F-1 score of 92 percent approx.
  Confusion matrix is as follows:

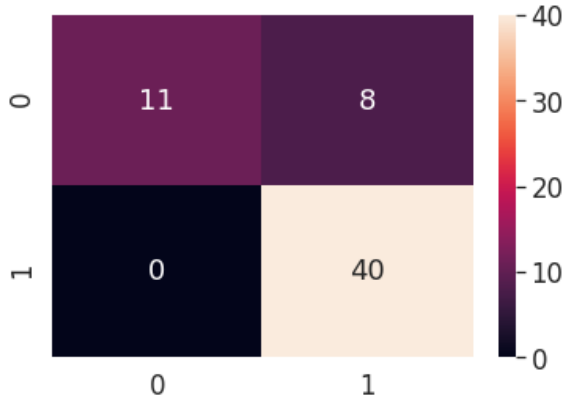

Figure 4

- Next model is with Neural Networks [7].
  We fed the data to the neural- networks and every time it gave an accuracy of 76 percent approx.
  F1- score of 83 percent.

Confusion Matrix as follows:

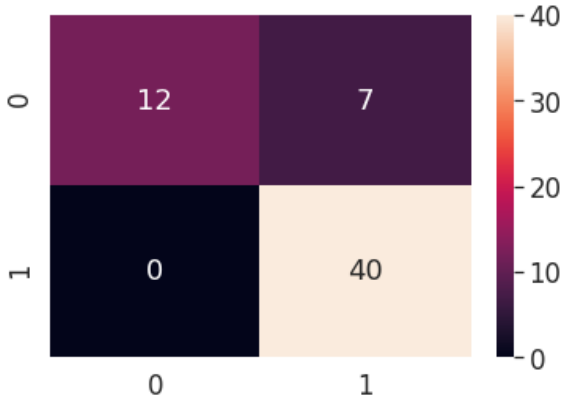

Figure 5

Loss (Figure 7) and Accuracy (Figure 7) curves for the DNN model:

Figure 6

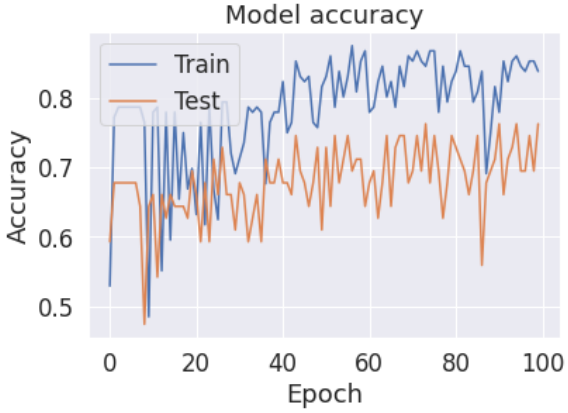

Figure 7

IV. RESULT AND DISCUSSION

With all the models we have seen that the best performing model is Random Forest and Decision Tree.

A. Further Discussion.

Neural Networks did not perform well with this dataset, the reason behind this may be the data of similar type i.e. all the data is vocal recording measurements. Neural networks try to find features inside the given features and that is not useful with similar kind of features. If a variety of features is present neural networks have performed better than this.

When we are training the data for long the training loss decreases and the validation loss is constant at the given time this shows that the model is getting overfitted. That is the reason why we have trained the model to 50 epochs only. We used dropout to create some generalization in the data but still, the accuracy was reliable.

Model comparison is given below:

Model	***Scores***
Decision Tree (Default) Accuracy,F1-score	*89, 92*
Decision Tree (max_depth=5, max_leaf_nodes= 2) Accuracy,F1-score	*81, 86*
Decision Tree (max_depth=5, max_leaf_nodes= 5) Accuracy, F1-score	*86, 90*
Decision Tree (max_depth=5, max_leaf_nodes=10) Accuracy, F1-score	*86, 90*
Decision Tree (max_depth=5, max_leaf_nodes=15) Accuracy, F1-score	*86, 90*
Random Forest (max_depth=5,max_leaf_nodes=15) Accuracy, F1-score	89, 92

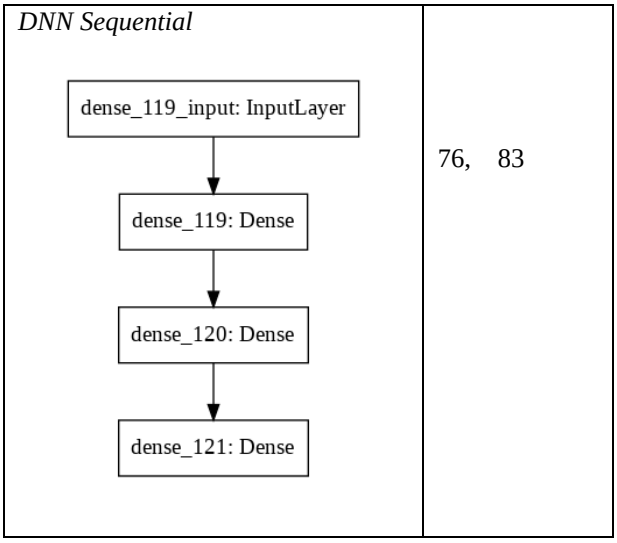

Figure 8

V. CONCLUSION AND RECOMMENDATION

With all the models the best score achieved is 92 percent with the decision tree classifier and Random Forest Classifier so it should be used in case of classification of PD with the dataset used.

It gives the best prediction that whether a person is suffering from Parkinson's Disease or not in an early stage.

VI. FUTURE WORKS

When the data is more complex in terms of adding more sensor data, Neural networks can perform better but as far as we are concerned with similar data with sensor information of similar types Classical Machine Learning Algorithms can perform better.

We can use synthetic data generator SMOTE [8] to reduce the class imbalance to further improve the scores. In a case where data is having more number of outlier, we can treat them using Z-score for better treatment of variance in the data as far as generalizing the model for better understanding the data.

VII. REFERENCE

[1] "Parkinson's Foundation: Better Lives. Together."

[2] Prof Thomas Davenport, "The potential for artificial intelligence in healthcare"

[3] “UCI Machine Learning Repository: Parkinson’s Data Set.”
[4] “Exploratory Data Analysis Wikipedia”
[5] “Application of Decision Tree Algorithm for Data Mining in Healthcare Operations: A Case Study”
[6] “A Random Forest based predictor for medical data classification using feature ranking”
[7] “Applications of artificial neural networks in health care organizational decision-making: A scoping review”
[8] “SMOTE for high-dimensional class-imbalanced data.”