Detection of Parkinson's Disease Through Telemonitoring and Machine Learning Classifiers

Arpan Adhikary[1], Koushik Majumder[1(✉)], Santanu Chatterjee[1], Anurag Dasgupta[2], Rabindra Nath Shaw[3], and Ankush Ghosh[3]

[1] Department of Computer Science and Engineering, Maulana Abul Kalam Azad University of Technology, West Bengal, Kolkata, India
`koushikwbutcse@gmail.com`
[2] Valdosta State University, Valdosta, GA 31698, USA
[3] University Center for Research and Development (UCRD), Chandigarh University, Mohali, Punjab, India

Abstract. Parkinson's Disease (PD) is one of the incurable neurodegenerative disorders. This progressive nervous system disorder mainly occurs at the age of early 60 and become worst day by day. Till now we do not have any particular medicine or surgery for this disease. As PD cannot be fully curable, so it is important to detect PD at its early stage to prevent more harm. PD detection is also important at its early stage as by the time of manifestation of clinical symptoms occur, more than 60% dopaminergic neurons lost by the time. Parkinson's Disease detection has now become a popular field of study and research. In this chapter, we use the audio medical measurement of 42 people, who are distinguished with early-stage Parkinson's Disease. The subjects were hired for a six months trial of a remote-controlled disease to remotely diagnose between PD and healthy people. The data were automatically recorded in subjects' homes. We have applied Linear Regression, Polynomial Regression, Elastic-Net, Lasso, Decision Tree, k-Nearest Neighbour, Random Forest and Gradient Boosting Regression on the dataset, to achieve the model's performance in terms of accuracy. This research will be useful in early diagnosis of Parkinson's disease and to prevent its' harmful impact on the patients.

Keywords: Parkinson's Disease (PD) · Machine learning · Linear regression · Polynomial regression · Elastic-net · Lasso · Decision tree · k-Nearest neighbour · Random forest and gradient boosting regression

1 Introduction

Parkinson's Disease (PD) is a common brain and nerve disorder. Generally, it affects the people mainly above 60 but depending upon some external scenarios, it can affect people of different ages. It can be genetic. People, exposed to pesticides and medicine like phenothiazine, reserpine etc. for a long period of time can also be affected by this disease

[1, 2]. People with PD meet some well-known symptoms i.e., slowness of movement, tremoring, rigidity, speech disorder etc. Some patients go through a pre-motor stage of 5–20 years before the occurrence of PD symptoms [3]. In this stage, patients face sleep behavioral disorder, rapid eye movement, olfactory loss, decreased sense of smell etc. Though any proper treatment for this disease is not discovered, but regular medication, meditation, physiotherapy can improve the patients' health to some extent [4, 5]. As speech disorder is one of the major symptoms of PD. Since examining speech disorder is easier, reliable and can be extensively adopted in telemonitoring process, it is now becoming a core area of research in Parkinson's Disease detection and prediction. In our chapter, we have predicted the UPDRS indicators from the sound measurements. Section 2 gives the detailed description about the related works done in this area; Sect. 3 gives the detailed information about our dataset. In the rest sections, we have discussed the detailed about data preparation, proposed methodology, results and discussion.

2   Related Works

In the recent past, ML and DL classifiers were taken into consideration by many researchers to classify Parkinson's Disease. Different features i.e., Freezing of GAIT (FoG), speech disorder data, walking data, handwritten image etc. were taken. Some of the research chapters are discussed in this section.

Pedro Gómez-Vilda1 et al. [6] in their study detect Parkinson's Disease from speech articulation. They have co-related the kinetic behaviour of patients' jaw-tongue biomedical system. They took the dataset of sustained vowels, which were recorded with the PD patients. The similar work was done for the normal people. They achieved 99.45% accuracy for Male patients and 99.42% accuracy for female patients. The sensitivity measurement was 99% for male patients and 99.42% for female patients.

Imanne El Maachi et al. [7] took gait sample as input to classify PD from healthy people. 1-D convnet neural network was used to build a DNN model. They have separated their work in two separate parts. First part contains 18 parallel 1D convnet to corresponding system inputs. In the second part of their work, PD was classified using Unified Parkinson's Disease Rating Scale (UPDRS) scale. They achieved 98.7% accuracy for their work.

Shivangi et al. [8] used deep neural networks for Parkinson's Disease detection. They introduced VGFR Spectrogram detector and voice impairment classifiers to diagnose PD at its early stage. CNN was implemented on gait signals, which were converted to spectrogram image. Deep dense ANN was implemented on voice recording to predict this disease. Using VGFR, they achieved 88.1% accuracy, while using Voice Impairment Classifier, they achieved 89.15% accuracy.

Gabriel Solana-Lavalle et al. [9] used kNN, MLP, SVM and RF on a small set of vocal features to detect PD. The main aim of this work was to increase the accuracy for PD detection and reduce the number of features for PD detection. Firstly, the speech samples were recorded from different individuals and processed for feature extraction. In the next stage, they have standardized to set the mean at 0. They achieved 94.7% of accuracy, 98.4% of sensitivity, 92.68% of specificity and 097.22% of precision for their work.

Tao Zhang et al. [10] shows different characteristics of voice signals between PD patents and healthy people. They used energy direction features which are based on Empirical mode decomposition. The work was done on two different datasets and achieved 96.54% and 92.59% accuracy repeatedly for both of the works. For classification, SM and RF classifiers were used on the extracted features and the best accuracy was calculated.

3   Data Collection

This is an open-source dataset, which is collected from UCI Machine Learning Repository [11]. The dataset first was created by Athanasios Tsanas and Max Little. Both of them are from the University of Oxford. They have collaborated with 10 medical centers from US. The telemonitoring device, which was used for data collection, was developed by Intel Corporation. This device was used to record the speech signals. The main objective was to predict the clinical Parkinson's Disease (PD) based on the Unified Parkinson's Disease Rating Scale (UPDRS).

4   Dataset Information

The dataset we take for our work contains the biomedical voice measurements of 42 people, who were already affected with early-stage PD. A six months trial was done using the telemonitoring device at the patients' home. The columns include an index number of each patient. Other columns include the patients' age, gender, duration from the initial date of admission, two motor indicators: motor UPDRS and total UPDRS. 16 other medical sound measurements were taken in the next 16 columns.

5   Data Pre-processing

The subject column in our dataset basically identifies each subject uniquely. But it is unordered and make no sense. It can also be confused for the classifiers while training the model. So, we can convert this column using one-hot encoding or discard it. But using one-hot encoding will add too many features to the dataset, then it will be hectic task to manage the features. So, the safest way is to remove the subject column from the dataset.

Our dataset does not contain any null values. So, we have directly divided the table into input and output elements. We have split the data in 7:3 ratio. 'SelectKBest' method was used to select the feature according to the k highest score. It takes the first k features which have the highest scores. As, we have around 200 data for each subject, so we have selected the best features for further processing.

To find the relation between two variables, co-relation measurement is the commonly used term. The most common co-relation measuring formula is Pearson co-relation. It assumes that the random variables of the dataset are distributed with a Gaussian normal distribution and linearly dependent on another variable. Linear co-relation is measured in the range of $(-1, 1)$, where 0 indicates no-relation between two variables and 1 indicates

highly co-relation between two variables. We have two variables motor UPDRS and total UPDRS as the random variables, so we checked the co-relation of other features with respect to these two features one by one (Figs. 1 and 2).
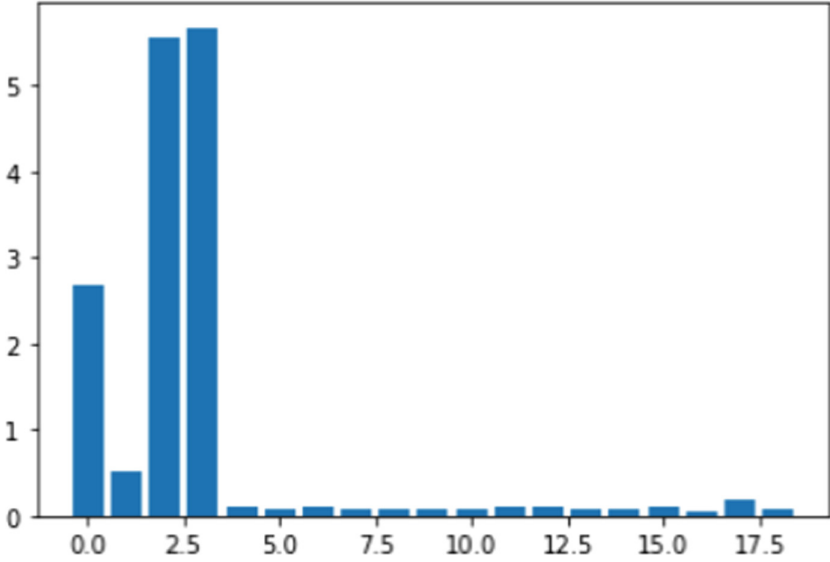

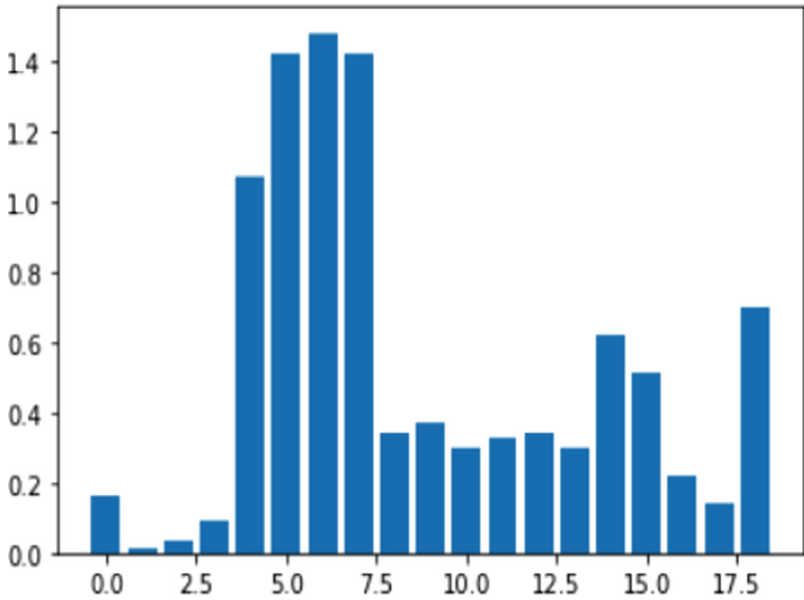**Fig. 1.** Measured co-relation of the features to motor-UPDRS

Fig. 2. Measured co-relation of the features to total-UPDRS

This heatmap shows the co-relation between the features. It is a graphical representation of the features, that shows the relation between the values. Here, we have co-related the initial features. Here, dark shades show high co-relation and white shows no co-relation between the columns (Fig. 3).
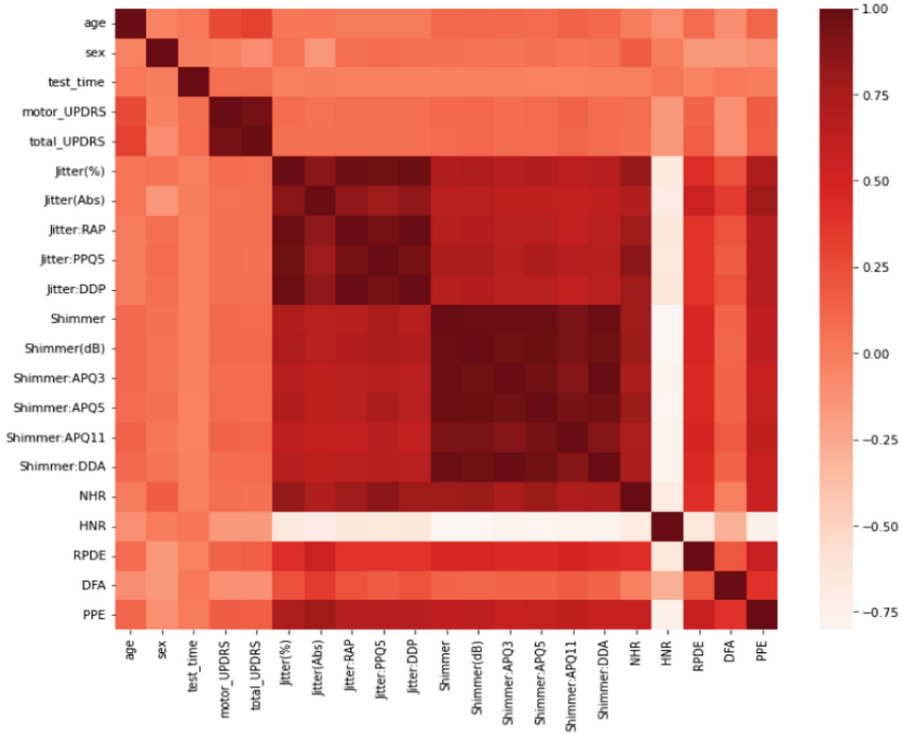
Fig. 3. Heatmap of the co-relation between features

Mutual information is the term, that is derived from information theory. It is calculated between two variables and measures the reduction of uncertainty for a random variable, where the value of another random variable is considered as known. Here, the k-best features are those features that have most mutual information. We have putted a threshold value (>0.1) to see which random variables are the most positively co-related with the random variable y, that we want to predict (Figs. 4 and 5).

Another method to check the co-relation is a machine learning classifier and test it repeatedly in subsets of our original data set, until we find which one makes the best prediction. It is a more expensive computing method, but it is more accurate than the previous ones. In this case, we have used the classifier ordinary least squares. Several techniques are there to do this,

```
age                0.273665
motor_UPDRS        1.000000
total_UPDRS        0.947231
Shimmer            0.102349
Shimmer(dB)        0.110076
Shimmer:APQ11      0.136560
HNR                0.157029
RPDE               0.128607
DFA                0.116242
PPE                0.162433
Name: motor_UPDRS, dtype: float64
```

Fig. 4. Measured most positive co-relation with motor-UPDRS

```
age                0.310290
motor_UPDRS        0.947231
total_UPDRS        1.000000
Shimmer:APQ11      0.120838
HNR                0.162117
RPDE               0.156897
DFA                0.113475
PPE                0.156195
Name: total_UPDRS, dtype: float64
```

Fig. 5. Measured most positive co-relation with total-UPDRS

o **Backward Elimination**
In backward elimination process, we started training the model with all the features and then started removing the features in every step. We have evaluated the performance of the algorithms through its matric p-value, which was set to 0.05. When the p-value is greater than 0.05, we removed the attribute, else we kept those.

o **Recursive Feature Elimination (RFE)**
In this process also we have removed the features until we reached to the best possible solution. But the main difference between backward elimination and RFE is, RFE uses the accuracy score. RFE takes the number of features as input and calculate the accuracy score. The output of RFE shows the ranking among the features.

6   Feature Selection and Model Training

After having all the previous work done, we have chosen a subset of most important features to work further. The subset of features was taken only based on the backward elimination.

Principal Component Analysis (PCA) is an unsupervised machine learning technique which is used to minimize the dimensionality of data. At the same time, it increases the

interpretability, and minimizes the information loss. It also helps to capture the most relevant features in the dataset. PCA has three steps. In the first step, it standardizes the range of the continuous initial features. Due to this, all the initial variables contribute equally to the analysis. After standardization, covariance of matrix computation is done to check the relationship among the input features. In order to determine the principal components of the features, in the next step, the eigen vectors and eigen values are computed. We have used PCA to find the best possible features that were extracted from the voice samples.

After feature extraction, we have used Linear Regression, Polynomial Regression, Elastic-Net, Decision Tree, k-Neighbor, Random Forest, Least Absolute Shrinkage and Selection Operator (Lasso), and Gradient Boosting to train our model.

Linear Regression

Linear Regression is a supervised machine learning model that finds the best fitted linear relationship between the independent and dependent variables. Linear Regression is of two types, Simple Linear Regression and Multiple Linear Regression. In Simple Linear Regression, only one dependent variable is possible. Whereas, in Multiple Linear Regression, more than one independent variable is possible. As, in our dataset, two dependent variables are there: motor UPDRS and total UPDRS, so we have used Multiple Linear Regression.

Suppose y is the dependent variable and b_0 is the intercept. b_1, b_2, ..., b_n are the coefficients of the independent variables x_1, x_2, ..., x_n. Then the equation will be,

$$y = b_0 + b_1x_1 + b_2x_2 + \cdots + b_nx_n$$

The main motive of Linear Regression is to find the best fitted linear line and optimal values of the intercept and coefficients to minimize the error. Here, error is the difference between the actual and predicted value.

Polynomial Regression

In polynomial regression, the relationship between the dependent and independent variables are described as the nth degree polynomial of the independent variable. It describes a non-linear relationship between the value of independent variables and the conditional mean of the dependent variable.

Suppose y is the dependent variable and b_0 is the intercept. b_1, b_2, ..., b_n are the coefficients of the independent variable x. Then the equation will be,

$$y = b_0 + b_1x_1^2 + b_2x_1^3 + \cdots + b_nx_1^n$$

Sometimes it is also called as a special case of the Multiple Linear Regression (MLR) as it adds some polynomial terms to the MLR equation.

This method is used to train the model in a non-linear manner.

Lasso

Lasso is a regression model based on the linear regression technique. It refers to the shrinking of the extreme values of the data sample towards the central values. This process makes lasso regression better, stable and less erroneous than others. It is considered

as one of the most suitable models for the scenarios having multi co-linearity. Lasso makes the regression method simpler in terms of the number of features which are used for the work. Lasso performs L1-regularization and the penalty added is equivalent to the magnitude of co-efficient. Lasso uses a regularization method which automatically penalize the extra features taken i.e., the features which are less co-related to the target variable.

Elastic Net Regression
Elastic Net regression is a linear regression that uses the penalties from lasso technique for regression model regularization. Elastic Net method performs regularization and variable selection simultaneously. It is the most appropriate for the scenarios where the dimensional data is larger than the number of samples. Variable selection and grouping play a major key role for elastic net techniques. It does not eliminate the high collinearity co-efficient.

Decision Tree
Decision Tree (DT) is a supervised machine learning algorithm, which is used for both classification and regression problems. DT is one of the predictive modelling approaches, that is used in data mining, statistics and machine learning. This algorithm split the input data into the sub-spaces based on some certain functionalities. It helps to reach to a conclusion based on some conditional control statement. But it is mostly used for classification problems. It is a tree-structured based classifier, in which each internal nodes represents each feature of the dataset. The branches of the tree represent the decision rules. Whereas, each leaf node of the tree represents the outcome of the algorithm. Decision nodes makes the decision rules. The goal of DT is to create a model, that can predict the value of the target variable by using simple decision rules inferred from the data features.

K-Nearest Neighbors
kNN is one of the supervised learning algorithms, that can also be used for both classification and regression problems. But it is generally used for classification problems. Here K is an important parameter for kNN. This algorithm counterfeits the similarity of the available data with the new data. Based on the data points' similarity, it puts the new data to the most similar category of the available data. Based on similarity, the algorithm easily classifies the new data points. kNN algorithm is robust to the noisy training samples and it is effective when training dataset is large. It is a non-parametric algorithm, that does not make any assumptions on the underlying data. It is a slow learner algorithm, as it does not learn from the training samples immediately. It stores the dataset and at classification process, it performs the necessary actions on the dataset.

Random Forest
RF is an ensemble learning algorithm, which is used for both classification and regression. It builds decision trees based on different samples. The algorithm takes their majority vote for classification. To build a RF, we need some DT, which must not be co-related or have low co-relation. RF can handle the dataset of continuous variables for regression

problems and categorical variables for classification problems. But this algorithm performs better for classification problems. RF combines multiple trees to predict a class of the dataset. So, it is possible that some of the DT predict the correct output. While the rest DT may not predict the correct output. But, together, they all can predict the correct output. So, there are two assumptions: the predictions of each tree must have very low co-relations and some actual values in the feature variable should be there, so that the classifier can predict accurate results. RF takes less training times and predict highly accurate output, even for large dataset.

Gradient Boosting Regression (GBR)
Gradient Boosting is one of the machine learning (ML) technique that is also used for both classification and regression problems. It produces a predictive machine learning model from an ensemble of weak predictive models. It is used to find the non-linear relationship between the dependent and the independent variables. This algorithm works better with missing values, outliers and high cardinality categorical values.

7   Result and Discussion

The main contribution of this work was the proposal of the subjects' voice signal data as input to the Machine Learning algorithm for PD detection. Employing public dataset, we evaluated the detection capability of PD from different machine learning classifiers. The most effective result we achieved using Decision Tree classifier. It gives us the accuracy of 100% on training dataset. Whereas, for control dataset, it gives us the accuracy of 97.05%. The algorithms used in our chapter represents a non-invasive and reliable methods for Parkinson's Disease detection. At first, we have taken the features to train our model, which are highly co-related with the dependent variables. Based on that we have created the dataframe for training and testing. Principle Component Analysis (PCA) was done for grid search with cross validation of the data. 'compute_metrics' function was used to calculate the matrices. Least square and mean square are also used to find the error on the training and control samples by the classifiers. 'GridSearchCV' is a technique, which searches through the best parameter values from the given grid of parameters. Grid search technique was used for lasso, elastic net, decision tree, gradient boosting and random forest to assemble the steps, which can be cross-validated together by setting different parameters. Below, we have compared between different classifiers used based on their accuracies on training and control set, errors in both of the samples and time taken to fit and predict the results (Table 1).

From the table, we can find, though Random Forest gave us better accuracy than the other algorithms, but it takes the most time to be fitted. Gradient Boosting also takes quite considerable time than others. As RF makes the decision based on different DTs, so for real-time prediction, it is slow.

In the below graph, we have compared between the training accuracy achieved by the classifiers that are used in our work. This barplot gives us a clear idea about the accuracy measurement. Figure 7. Shows the running time measurement of different algorithms. In this figure we can clearly see that RF and BGR took very high time than the other algorithms (Fig. 7).

Table 1. Comparing different classifiers based on the results.

Classifier name	Accuracy on training data set	Accuracy on control data set	Average square error in training samples	Mean square error in control samples	Time to fit and predict (seconds)
Linear regression	93.79%	93.95%	5.2793	5.4187	0.018
Polynomial regression	96.55%	92.12%	3.1680	3.5858	0.138
LASSO	92.57%	93.48%	5.4880	5.4530	1.863
Elastic net	93.79%	93.95%	5.2793	5.4187	14.745
Decision tree	100.00%	97.05%	0.0000	0.2189	0.231
k-Nearest neighbors	89.71%	79.10%	0.5586	1.7661	0.156
Random forest	99.73%	98.63%	0.0150	0.0973	12.527
Gradient boosting regressor	98.63%	97.74%	0.9905	1.0945	5.538
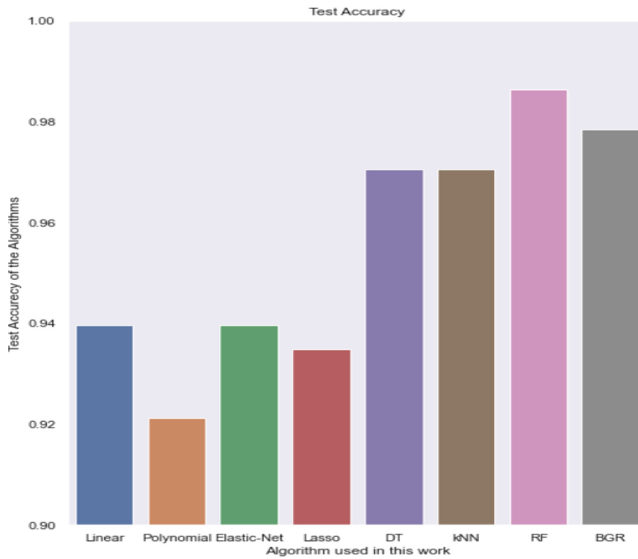
Fig. 6. Training accuracy measurement of different classifiers
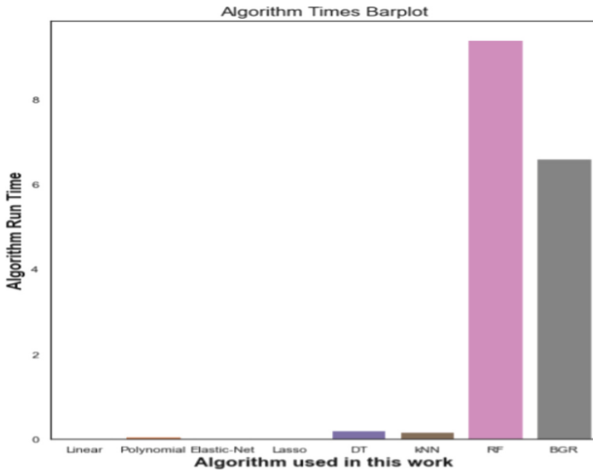
Fig. 7. Running times of the algorithms

Algorithm tuning time is an important parameter for model selection. It is used to check, how much time the algorithms take to be trained using the dataset. Different classifiers take different times based on the features added to the dataset. Here in this bar plot, we showed the comparison of the classifiers that are used in this work (Fig. 8).
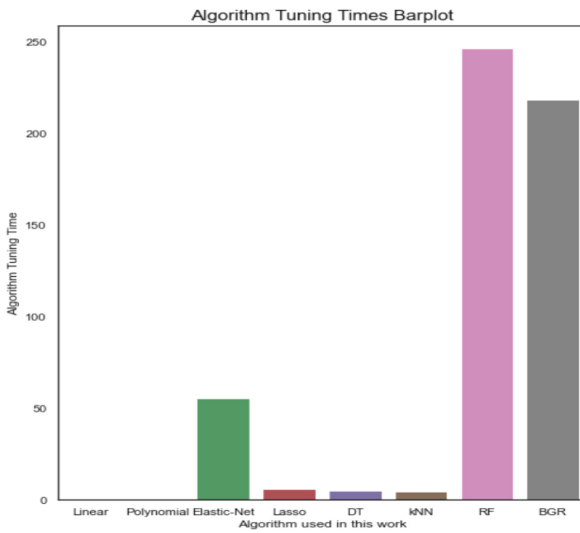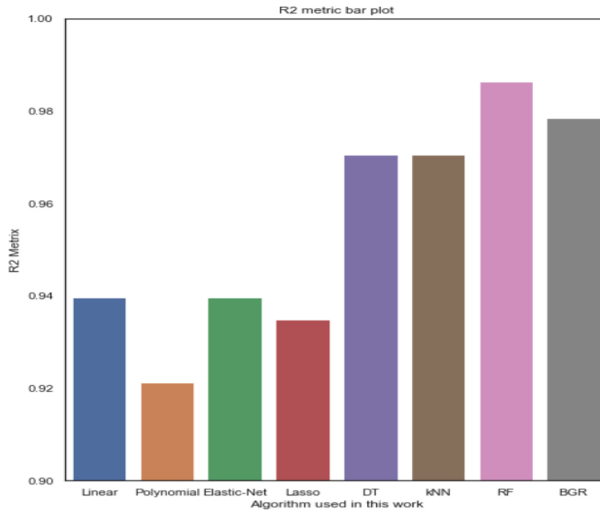

Fig. 8. Tuning time taken by the algorithms

R^2 is a statistical measure of fit which indicates the portion of dependent variable which is explained by the independent variable in a regression model. We have taken R^2 measure for our work (Fig. 9).

Fig. 9. R^2 metrics measure for the algorithms

Mean Square method is another technique, that is used to measure the regression models' performance. It takes the distance between the data points and the regression line to squares them removing negative sign. It gives more weightage to the larger difference. Smaller MSE shows the best fitted line (Fig. 10).
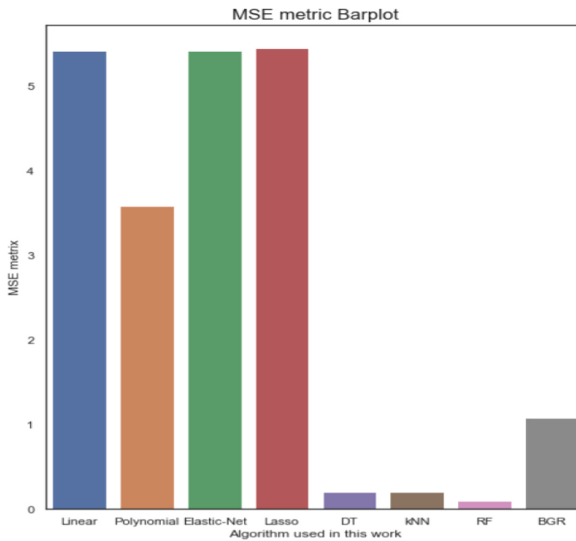
Fig. 10. MSE measure for the algorithms

8 Conclusion and Future Scope

PD is a nerve disorder, that challenges the population globally as it is uncertain in prediction. In our dataset, we have used the audio measurements of 42 peoples, who were already classified with early Parkinson's Disease. The patients were sent for a six-months trial where they were remotely diagnosed and the data were automatically recorded through the device. Total 5,875 different sounds were measured for the test. UPDRS is a rating tool, that is used to gauge and progression of PD in the patients. This scale has been modified over the years. Several medical organizations have modified UPDRS and continues to be one of the bases for research and treatment for PD. In this work, we have taken both motor UPDRS and total UPDRS score for PD detection. Using Linear Regression, we achieve 93.79% for training data. Whereas, for control data, we achieve 93.95% of accuracy. Polynomial Regression gives us 96.55% and 92.12% of accuracy for both training and control data. The best accuracy we achieved using Decision Tree classifier. It gives us the accuracy of 100% and 97.05% for both training and control data. Though we have achieved better accuracies using the classifier used, but the tuning time for Elastic-Net, RF and BGR was bit longer. So, in future, it can be tested further with the other algorithms. Also, the best fitted algorithm: both time wise and complexity wise can be taken for further process. After UPDRS score, we will classify PD patients from normal people and progression of PD.

References

1. Pahuja, G., Nagabhushan, T.N.: A comparative study of existing machine learning approaches for Parkinson's disease detection. IETE J. Res. **67**(1), 4–14 (2021)
2. Shi, T., Sun, X., Xia, Z., Chen, L., Liu, J.: Fall detection algorithm based on triaxial accelerometer and magnetometer. Eng. Lett. **24**(2) (2016)
3. Ali, L., Zhu, C., Golilarz, N.A., Javeed, A., Zhou, M., Liu, Y.: Reliable Parkinson's disease detection by analyzing handwritten drawings: construction of an unbiased cascaded learning system based on feature selection and adaptive boosting model. IEEE Access **7**, 116480–116489 (2019)
4. Abdulhay, E., Arunkumar, N., Narasimhan, K., Vellaiappan, E., Venkatraman, V.: Gait and tremor investigation using machine learning techniques for the diagnosis of Parkinson disease. Future Gener. Comput. Syst. **83**, 366–373 (2018)
5. Wang, W., Lee, J., Harrou, F., Sun, Y.: Early detection of Parkinson's disease using deep learning and machine learning. IEEE Access **8**, 147635–147646 (2020)
6. Gómez-Vilda, P., et al.: Parkinson disease detection from speech articulation neuromechanics. Front. Neuroinform. **11**, 56 (2017)
7. Palimkar, P., Shaw, R.N., Ghosh, A.: Machine learning technique to prognosis diabetes disease: random forest classifier approach. In: Bianchini, M., Piuri, V., Das, S., Shaw, R.N. (eds.) Advanced Computing and Intelligent Technologies. LNNS, vol. 218, pp. 219–244. Springer, Singapore (2022). https://doi.org/10.1007/978-981-16-2164-2_19
8. Johri, A., Tripathi, A.: Parkinson disease detection using deep neural networks. In: 2019 Twelfth International Conference on Contemporary Computing (IC3), pp. 1–4. IEEE (2019)

9. Mridha, K., et.al.: Plant disease detection using web application by neural network. In: 2021 IEEE 6th International Conference on Computing, Communication and Automation (ICCCA), pp. 130–136 (2021). https://doi.org/10.1109/ICCCA52192.2021.9666354

10. Zhang, T., Zhang, Y., Sun, H., Shan, H.: Parkinson disease detection using energy direction features based on EMD from voice signal. Biocybern. Biomed. Eng. **41**(1), 127–141 (2021)

11. Dataset used for this work. https://archive.ics.uci.edu/ml/datasets/parkinsons+telemonitoring