Parkinson's Disease Prediction using Modified Gauss-Newton Method in Feed-Forward Neural Network

Madhuri Gupta, Bharat Gupta

Abstract*: *Parkinson's disease (PD) is a brain disorder, characterized by the relapse of the nervous system that spreads gradually in the body. The symptom of PD includes a loss of body control (moderate movement, resting tremors, postural shakiness etc.). So, it is required to detect at an early stage. Machine learning (ML) deals with a variety of probabilistic methods to identify a pattern in a dataset. Therefore, the research is carried out to predict the PD using Multilayer Feed-Forward Neural Network. In Neural Network (NN), weight optimization performed at each layer that plays a major role in the prediction. First-order weight optimization techniques are slow in computation because they reduce the sum of square error using parameter updating in the steepest descent way. In proposed work, a modified recursive Gauss-Newton method is used to optimize the weights for speed up the performance of Feed-Forward NN. This approach is compared with widely used optimization techniques. The Proposed method found better than other techniques and performs fast in Apache Spark than R-Studio framework.

Keywords*: *Apache Spark, Feed-Forward Neural Network, Gauss-Newton Method, Gradient Descent, Machine learning, Parkinson's disease.

I. INTRODUCTION

As per Parkinson's foundation [1], Parkinson's disease diagnosed 60,000 patients each year from approximately 10 lakh Americans. Worldwide more than 10 million people are suffering from this disease. Parkinson's disease [3] comes under group of neurological disorders that affect movement problems such as tremors, slow movement and stiffness. A typical Parkinson's disease is also called Parkinsonism. According to Figure 1, Parkinson's disease affects 10 lakhs Americans with 60,000 diagnosed each year that shows the advancement is needed for early parkinsonism detection. Prediction of this disease is possible by exploiting the information from the dataset of the disease. The purpose of the research work is to improve the prediction of Parkinson's disease.

In the research, machine learning techniques are used for the prediction of Parkinson's disease. Application of ML in the healthcare field is growing rapidly due to the efficiency of its approach [3], [4]. Multi-layer feed-forward neural network is applied here to predict the disease. In neural network, weight optimization plays vital role at each layer. During the training phase of the NN, learning algorithm adjusts the input parameter like weights and bias so that the model can predict the outcome with higher accuracy and in minimum timeframe. First-order weight optimization techniques are the slogger from past years but these

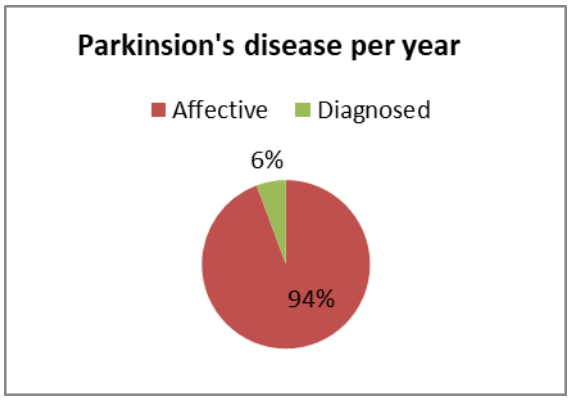

Fig. 1: Graphical representation of Parkinson's disease per year [5]

Techniques are slow in computation because they reduce the sum of square error (SSE) using parameter update in steepest decent direction. Eventually, first–order techniques stab to passes towards the local optimum resolution by slowly moving down the curve. Whereas second order method, reduce SSE by assuming the least square function as a quadratic (local) than find the minimum of quadratic function.

In the research work, feed forward Neural Network is used with recursive Gauss-Newton (second-order) method with respect to accuracy, R-Square, F-score and 10-fold cross validation. Experiments are conducted on Apache spark data processing engine. NN is constructed on TensorFlow and visualize using TensorBoard in python programming language. This experiment also performed on R-Studio platform using R- Programming language.

The paper arranges in five sections; next section explained related works. Method section describes the classification techniques. Experimental design section describes about the dataset, framework, software, pre-processing and feature extraction techniques used. Result and discussion section discussed about evaluation techniques and to the journal, rectification is not possible.

Revised Manuscript Received on November 08, 2019.
***** Correspondence Author
Madhuri Gupta*, Computer Engineering and Information Technology, ABES Engineering College, Ghaziabad, India. Email: madhuri.gupta@abes.ac.in.
Bharat Gupta, Computer Science and Information Technology, Jaypee Institute of Information Technology, Noida, India. Email: bharat.gupta@jiit.ac.in.

II. RELATED WORK

This section explains the research work carried related to speech Parkinson's disease prediction using machine learning techniques. Guo, Fang et al. [5] combine the genetic programming technique and expectation maximization to generate learning feature functions of Parkinson's data using basis feature of voice data. By using expectation maximization, the transformed data generated as a Gaussians mixture.

Sriram and Rao et al. [6] used machine learning algorithm to detect Parkinson's disease and results shows Support Vector Machine has better accuracy (88.9%) in contrast to K-NN and Naïve Bayes technique that shown accuracy 69.23%. Hierarchal clustering and SOM techniques predicted higher number of clusters in healthy dataset and less number in diseased data. According to Tsanas et al. [7] speech is a beneficial signal for perceptive PWP from healthy panels [8], [9]. In fact, vocal diminishing is the initial prodromal Parkinson's disease symptoms, predictable up to five years before the clinical diagnosis [10]. Table 1 shows the survey report of Parkinson disease.

Aich et al. [11] performed Principal Component Analysis (PCA) on different features set of gait and voice parkinson's datasets then applied a nonlinear tree-based classification technique which results in 96% prediction accuracy. Nilashi et al. [12] proposed a hybrid model of SVM and SOM (Self Organizing Map) machine learning techniques for parkinson's disease prediction. The parameter of prediction is Mean Absolute Error (MAE) for the Motor-UPDRS and Total-UPDRS that results correspondingly MAE = 0.4967 and MAE = 0.4656. The study shows that machine learning plays a vital role in prediction.

According to some research [13], [14], [15] speech is the strong evidence to discriminate the symptom of Parkinson from healthy subjects. Cooperatively, these results emphasize that speech is significant symptom to extract disease status, after processing of the recorded speech signals. Tsanas et al. [16], followed 1,140 individuals without loss of memory at starting point for a mean of 5.5 years.

As per related work, Parkinson's disease can predict by using some widely used machine learning techniques. Neural network and SVM works better to improve the accuracy of prediction. Parkinson Disease usually produces no symptoms at early stage. Therefore, some advance prediction techniques are required to predict the disease at its early stage.

III. METHOD

Parkinson's disease is rapidly increasing around worldwide [1]. In the research work, a model is proposed to predict the Parkinson's disease using feed-forward neural network with recursive Gauss-Newton method then compare it with gradient decent and some most extensively used machine learning techniques such as Support Vector Machine (SVM) and Logistic Regression (LR).

Table- I: Survey Report of Parkinson's Disease

S. No.	Machine Learning Techniques	Solution	Results
1	Poisson Regression [17]	meta-analysis of M–F ratios	95% confidence interval
2	Parallel Random Forest Technique [18]	Big Data analytics in a Spark Cloud Computing	0.2 error rate for 500 trees.
3	SVM, KNN, Random forest (RF) and Naïve Bayes [6]	Diagnosis of Parkinson disease	RF performed better with accuracy (90.26)
4	SVM Ensemble Model [19]	Parkinson disease diagnosis using computer-aided	97% accuracy with Rotation Forest
5	Neural Networks, Regression, DMneural and Decision Tree [20]	Prediction of Parkinson disease	Neural network performs better with accuracy 92.9%
6	K-NN, NB, Decision Tree [21]	Early stage prediction of Parkinson decision.	NB shows 83%, AD Tree shows 100% result.
7	Ensemble SVM classifier and random forest [22]	Improve the prediction accuracy of different disease.	SVM-RFE and ESVM-RFE result as 0.79and 0.88, respectively
8	SLSQP and Neural Network [23]	Weight assignment and Prediction	Class F-scores is 76.19% and 83.54% accurate respectively.
9	Logistic Regression, Decision Trees, and Factorization machines and Stacked Ensemble Learning [24]	Improve the performance of standalone classifiers using genomic data.	best model is average gradient boosting with Test ASE 0.09273
10	Artificial Neural network [25]	Prediction of Blood Glucose concentration	Feature based Neural Network has 91.43% of correct prediction.

A. Feed-Forward Neural Network

Feed-Forward Neural Network is a type of artificial neural network that does not form a cycle [26], [27], [28]. In this Neural Network, data flows in only one direction. Simplest feed-forward neural network is single-layer perceptron that contains a single layer therefore, this NN was not capable to produce the X-OR operation. So, Recursive Gauss-Newton method implied with multilayer feed-forward NN to overcome the limitation of single layer. Multilayer-Perceptron is capable to solve every Boolean function. In multilayer perceptron, information passes from input to hidden layer and hidden to output layer and connection between the neurons do not make cycle. In this research work 3 layers are used to predict the Parkinson disease. Weights are calculated by Recursive Gauss–Newton algorithm. Figure 2 shows the proposed approach by using three-layer feed-forward neural networks where exponential Linear Unit (ELU) [29] activation function is used at

hidden layer; Sigmoid Activation function is used at final layer and cross-entropy loss function is used to examine loss in network.

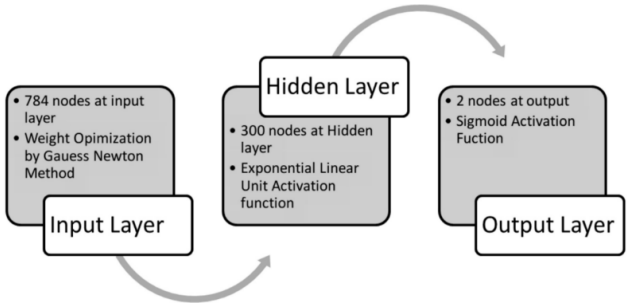

Fig 2: Proposed Approach

- *Input layer:* Number of neurons at input layer is same as the number of attributes in dataset. Here 1 bias and 26 attributes are applied as a predictor so, input layer containing 27 nodes.
- *Hidden Layer:* This layer lies between input and output layer and applies activation function on results, before passing to the nest layer. Here hidden nodes (HN) are calculated by Equation 1, which does not result in over-fitting.

$$H_N = S \Big/ (\delta * (I_N + O_N)) \qquad (1)$$

In Equation 1, S represents total number of samples presents in dataset, I_N represents number of nodes at input node and O_N represents number of nodes at output layer. Here δ, is a scaling factor that can be from 2 to 10, in this work, δ = 2 so 18 hidden nods are used.

- *Output Layer:* This is a final layer in the neural network. In the work, binary classification [30] is applied to resolve the diagnosis problem so one node is used at output layer. This layer receives input from previous hidden layer and applies activation and loss function then returns the model's prediction.
- *Activation Function:* In this work, Exponential Linear Unit (ELU) [31] is applied as activation function to speed up the learning and avoid bias shift because ELU is faster to tend the convergence cost to zero and give more accurate output. Zero is to increase the learning by bringing the gradient close to natural gradient. In comparison to ReLUs (Rectified linear units), ELUs provides negative results due to which, unit activations come closer to zero.
- *Loss Function:* In the proposed work, Cross-entropy loss function is applied to measures the performance of a classification model (Cross-Entropy Loss Function, 2018). It examines loss value between a "model" distribution of predicted class p(x) and a "true" distribution of discrete class d(y). Cross-entropy loss always results a probability value in the range of 0 to 1. In this function, Loss increases when the predicted probability of class is different from the actual class whereas an accurate classification model would have a cross-entropy loss to zero. Mathematically, in binary classification model, this loss is calculated as:

$$L(p_x, d_y) = -\sum_{A=1}^{E} d(y = E) \log p(x = E) \qquad (2)$$

In, Equation (2), loss value is based on A and E divergence between d and p.

B. Gauss–Newton Algorithm

Gauss-Newton is an iterative method to minimize the sum of square function value. Here iterative means, it solves series of calculation to find the optimum solution. This method is mostly used to minimize the second order approximation of the function. Basically, this algorithm is used to solve the non-linear least square problems. Gauss-Newton is generally used to find the best fit hypothetical model even if it can locate a single point.

Gauss-Newton's method is implied with neural network because it can offer a guarantee for its affine invariant, unique solution and most of all it converges in fewer steps.

This algorithm finds the value of the variables; iteratively that minimizes the sum of squares (3)

$$\delta\,(A) = \sum_{i=1}^{m} ri^2(A) \qquad (3)$$

In Equation 3, A is A (1, 2…n) for n variable and r is residual. Gauss-Newton method finds a solution for δ which used to minimize the squared function value, Starting with an initial parameter δ (0),

The updated step of Gauss-Newton and gradient decent method for an equivalent problem is follows

- *Gradient Decent*

$$\delta f^{(A+1)} = \delta^{(A)} - \mu\,\Delta\left(\frac{1}{2}\, r\,(\delta^{(A)})^T\, r(\delta^{(A)})\right) = \delta^{(A)} - \mu\, J_A^T r\,(\delta^{(A)}) \qquad (4)$$

- *Gauss-Newton*

$$\delta^{(A+1)} = \delta^{(A)} - (J_A^T J_A)^{-1}.\, J_A^T\, r\left(\delta^{(A)}\right) \qquad (5)$$

In Equation (4) and (5), r and δ are column vector and T represents the transpose of matrix and J is the Jacobean Matrix which can represent as:

$$(J_A)_i = \frac{\delta\, r_i\,(A)}{\delta A_j} \qquad (6)$$

$$J(A) = \begin{bmatrix} \frac{\delta r_1(A)}{\delta A_1} & \cdots & \frac{\delta r_1(A)}{\delta A_n} \\ \frac{\delta r_2(A)}{\delta A_1} & \cdots & \frac{\delta r_2(A)}{\delta A_n} \\ \vdots & \cdots & \vdots \\ \frac{\delta r_m(A)}{\delta A_1} & \cdots & \frac{\delta r_m(A)}{\delta A_n} \end{bmatrix}$$

Gradient Descent method required the gradient whereas hessian is required for Newton's Method. Every iteration of Gauss-Newton's method required to linear solve on hessian method.

- *Hessian*

The problem assists the approximation of the hessian method castoff with Newton's method using chain rule:

$$\delta^2 f(A) = \delta r(A) \delta r(A)^T + \sum_{i=1}^{m} r_i(A) \delta^2 r_i(A)$$

$$= J(A)^T J(A) + S(A) \qquad (7)$$

Thus, Hessian is a sum of two terms: $J_A^T J_A$ with first-order derivative and S (A) for second-order derivative

$$\text{H} = \delta\,^2 f(A) \approx J_A^T J_A + S(A)$$

Hessian makes the newton's method faster by minimize the second order Tylor-approx with $\delta(A)$ in each step. This second order information deals with convergence in a direct path.

- *Proposed Approach*

This section describes that how feed-forward network recursively solved by Hessian blocks:

Feed-forward NN take input vector as x0=a and create an output vector R_F on final layer R^{th} of neural network.

$$R_\lambda = w_\lambda\, x_{(\lambda-1)} \qquad (8)$$
$$x_\lambda = f_\lambda\,(R_\lambda)$$

Here λ lies between 1 and R, w_λ is a weight matrix, R_λ is pre-activation at λ layer. f_λ Is transfer function for each element and x_λ are activation values. Loss of neural network is examined by equation (2) L (p_x, d_y) between output p(x) and the actual output d(y). Total error of a network is calculated by the expected loss L (α).

$$\text{L}\,(\alpha) = \mathbb{L}\,[L]_{\,p_x, d_y} \qquad (9)$$

Here (a, b) is a single data point and α is:

$$\alpha = [v(w1^T), v(w2^T), v\,(w3^T) \ldots \ldots \ldots, v(w_R^T)]^{\text{T}}$$

v: vector

This research work is focused on the parameter Hessian Ħ. It became more optimized by combining with the loss function:

$$[\text{H}]_f = \left(\partial^2 f \Big/ \partial_i\, \partial_j\right) L(\alpha) \qquad (10)$$

Expression of this equation is used to find the expected parameter of hessian [32]. Here Hessian matrix is approximate by including loss of each layer in (10) as follows:

$$[\text{H}]_{ij} = \begin{bmatrix} \dfrac{\partial^2 f}{\partial \alpha \text{i}^2} & \dfrac{\partial^2 f}{\partial \alpha_i\, \partial \alpha_j} & \cdots \\ \dfrac{\partial^2 f}{\partial \alpha_j\, \partial \alpha_x} & \dfrac{\partial^2 f}{\partial \alpha \text{j}^2} & \cdots \\ \dfrac{\partial^2 f}{\partial \alpha_k\, \partial \alpha_x} & \dfrac{\partial^2 f}{\partial \alpha_k\, \partial \alpha_y} & \cdots \\ \vdots & \vdots & \ddots \end{bmatrix} \qquad (11)$$

Now, 'Only Hessian' $(\text{h}\upsilon_\lambda)$ is used to refer hessian for single data point (a, b) for discriminate with expected hessian. Only Hessian is computed by pre-activation that is calculated recursively as follows:

$$\text{h}\upsilon_\lambda = \frac{\partial}{\partial R_y^\lambda}\frac{\partial}{\partial R_x^\lambda} = \frac{\partial}{\partial R_y^\lambda}\sum_i \frac{\partial \text{L}}{\partial R_i^{\lambda+1}}\frac{\partial R_i^{\lambda+1}}{\partial R_x^\lambda}$$

$$= \sum_i \frac{\partial}{\partial R_y^\lambda}\left(\frac{\partial \text{L}}{\partial R_i^{\lambda+1}}\frac{\partial R_i^{\lambda+1}}{\partial R_x^\lambda}\frac{\partial R_x^\lambda}{\partial R_x^\lambda}\right)$$

$$= \sum_i w_{i,x}^{\lambda+1}\,\frac{\partial}{\partial R_y^\lambda}\left(\frac{\partial \text{L}}{\partial R_i^{\lambda+1}}\frac{\partial R_x^\lambda}{\partial R_x^\lambda}\right)$$

$$= \sum_i w_{i,x}^{\lambda+1}\left(\frac{\partial R_x^\lambda}{\partial R_x^\lambda}\,\frac{\partial^2 L}{\partial R_y^\lambda\,\partial R_i^{\lambda+1}} + \frac{\partial \text{L}}{\partial R_i^{\lambda+1}}\,\frac{\partial^2 x_x^\lambda}{\partial R_x^\lambda\,\partial R_y^\lambda}\right)$$

$$= \delta_{\,x,y}\frac{\partial^2 x_x^\lambda}{\partial^2 x_x^\lambda}\left(\sum_i w_{i,x}^{\lambda+1}\,\frac{\partial \text{L}}{\partial R_i^{\lambda+1}}\right) +$$

$$\sum_{i,j} w_{i,x}^{\lambda+1}\,\frac{\partial R_x^\lambda}{\partial R_x^\lambda}\,\frac{\partial^2 L}{\partial R_j^{\lambda+1}\,\partial R_i^{\lambda+1}}\,w_{i,y}^{\lambda+1}\,\frac{\partial R_y^\lambda}{\partial R_y^\lambda}$$

$$= \delta_{\,x,y}\frac{\partial^2 x_x^\lambda}{\partial^2 x_x^\lambda}\frac{\partial \text{L}}{\partial R_x^\lambda} +$$

$$\sum_{i,j} w_{i,x}^{\lambda+1}\,\frac{\partial R_x^\lambda}{\partial R_x^\lambda}\,\frac{\partial^2 L}{\partial R_j^{\lambda+1}\,\partial R_i^{\lambda+1}}\,\frac{\partial R_x^\lambda}{\partial R_x^\lambda}\,w_{i,y}^{\lambda+1} \qquad (12)$$

Henceforth, pre- activation for hessian is written as:

$$\text{h}\upsilon_\lambda = Y_\lambda\, w_{\lambda+1}^T \text{h}\upsilon_{\lambda+1}\, w_{\lambda+1} Y_\lambda + K_\lambda \qquad (13)$$

Here, Y_λ and K_λ are two diagonal matrices.

Now off-diagonal elements of eq. (13) based on (11) are ignored to simplify the complexity of method. Here, off diagonal elements of hessian matrix are set to zero (11) then apply pre-activation of only hessian. As a result, Y_λ and K_λ are identified as

$$K_\lambda = Diag\left(d'_\lambda(R_\lambda)\right) \qquad (14)$$
$$Y_\lambda = Diag\left(\; d''_\lambda\,\frac{\partial \text{L}}{\partial_{x\lambda}}\right) \qquad (15)$$

Here d' is first-derivative and d'' is second derivative of f_λ.

So, in above equation (13), $\text{h}\upsilon_\lambda$ is a recursive method of single data point that depends on expected loss function (L(α)) and diagonal elements. This recursion can simply use to compute the hessian activation for each layer of feed-forward neural network and update weights and bias for each layer.

Here, weight is estimated as:

$$w_{i,j}^{n+1} = w_{i,j}^n - \text{þ}.\frac{d\text{h}\upsilon'}{dw_{i,j}^n} \qquad (16)$$

Bias for each layer is estimated as:

$$b_{i,j}^{n+1} = b_{i,j}^n - \text{þ}.\frac{d\text{h}\upsilon'}{db_{i,j}^n} \qquad (17)$$

Here þ is stable learning rate.

C. Logistic Regression

Logistic regression (LR) is a technique of machine learning taken from statistics field [33, 34]. It is a linear model but predictions used to calculate by logistic function. Logistic function is basically sigmoid function, it takes any real number value and normalize it between 0 and 1, but never precisely at those limits. It is basically applied for binary classification problem in which dataset contains two classes. LR model measures probability of default classes.

$$p = {}^{1}/_{1 + e^{-value}} \quad (18)$$

In eq. (18), e is the base of the natural logarithm and value is the real numerical value which needs to transform.

D. Support Vector Machine

Another Classification technique used in this work is Support Vector Machine. It is supervised Ml technique with associated learning algorithm [35, 36,]. This technique is efficient due to its classification enactment. SVM technique has a decision hyper plane which divides the data sample of different classes on the basis of maximum margin. All those data samples, that are adjacent to hyper plane are called support vector. In the work Gaussian Radial Basis Function (RBF) is applied as a kernel function.

In this proposed work a comparative analysis of feed forward neural network is placed with Gradient Decent, SVM and logistic regression to improve the prediction of Parkinson's disease.

IV. EXPERIMENTAL DESIGN

This experimentation is evaluating the performance of feed forward neural network, along with this compare the performance with first-order technique and traditional ML techniques in terms of R-Square, accuracy and f-score to predict the Parkinson's disease.

A. Dataset

'Parkinson Speech Dataset' is used in this experiment. This dataset is available on UCI machine learning repository with various Sound Recordings Data [37]. The dimension of dataset is 1040×29 where 1040 represent the samples and 28 represent the attributes of data.

The Parkinson's disease database contains training as well as test files. Training data has 20 healthy individuals (10 males, 10 females) and 20 PWP (14 males, 6 female). Numerous sound recordings (26 voice samples including words, vowels, short sentences and sustained numbers) are taken from all patients. A group of 26 features on the basis of time and linear frequency are taken from every data pint. This data set is taken by Unified Parkinson's disease Rating Scale (UPDRS), that helps to rate the status of each patient who is determined by expert physician.

This dataset contains 28 attributes separated by commas for each data point.

- Feature 1: It indicates sample id.
- Features 2 to 27 indicates the features
- Features 1 to 5 shows Jitter (ddp), Jitter (rap), Jitter (ppq5), Jitter (rap), Jitter (local, absolute), and Jitter (local).
- Features 6 to 11 indicate shimmer (apq3), Shimmer (apq5), Shimmer (local, dB), Shimmer (dda), Shimmer (local), and Shimmer (apq11).
- Features 12 to 14 indicate NTH, HTN, AC,
- Features 15 to 19 indicate median, standard, deviation, mean pitch, maximum pitch and minimum pitch.
- Features 20 to 23 indicate number of periods, Number of pulses, Standard deviation of period, Mean period.
- Features 24 to 26 indicate fraction of locally unvoiced frames, Degree of voice breaks, Number of voice breaks.
- Attribute 28 indicates UPDRS (A rating tool that rate on the basis of longitudinal course of Parkinson's disease).
- Attribute 29 indicates class information.

Attribute 2 to 27 are used to form the feature vector. This vector is used to categorize between cancerous between Parkinson's disease and non-Parkinson's disease.

Table- II: Description of Dataset

Data Set Type	Multivariate (Sound Recordings)	Attribute Characteristic	Real
Associated Task	Classification	Number of instances	1040
Number of classes	2	Number of attributes	29

Table 2 shows the description of whole dataset in term of dataset characteristics, Attribute Characteristics, Number of Instances and Number of classes.

B. Performance Metrics

The performance of this experiment is analyzed on the basis of some parameters such as R-square, Accuracy and F-score. 10-fold cross validation is applied to validate the outcomes:

- *R2 –Score (Coefficient of determination):* It is defined as the measure of how close the data points are replicated by the regression line [38]. It ranges from 0-1.
- *Accuracy:* It is defined to measure the closeness of a classification model to the original value [39].
- *F-Score:* It is calculated as a harmonic mean of precision and recall [40].
- *Cross validation:* Cross-validation [41] is a technique to validate the performance of classification model. It uses some computation method instead of mathematical exploration. In this technique original dataset divides into K parts. Among them K-1 data parts are used for training set to train the classification model and 1 data part is used as test set for evaluating the model. In this proposed work, 10-fold cross validation is applied to evaluate the classification model.

C. Proposed Model

In the research work, a classification model is proposed to predict the Parkinson's disease using three-layer feed-forward neural network incorporate with Recursive Gauss-Newton method. Fig. 3 shows the flow diagram of proposed experiment. Proposed model is compared with most extensively used First-order optimization method and machine learning technique on Parkinson Speech dataset. The work flow of this proposed work is as follows:

- *Step 1:* Dataset is divided in two parts; one part is training data and other is testing data. Training data is used to train the model by using label containing the class values. Test data applied on classification model to predict the disease on the basis of outcome.
- *Step 2*: Learning Model In this step, prediction model is generated using Feed-forward neural network by applying gauss-newton method for weight estimation at each layer. This method is used to handle the mean square function value of dataset which results in improved accuracy. Three ML techniques LR and SVM are used to train data parts.

Afterward, test data applied on proposed model and compare their results by performance parameter. This technique generated the final outcome that the disease is Parkinson or non-Parkinson.

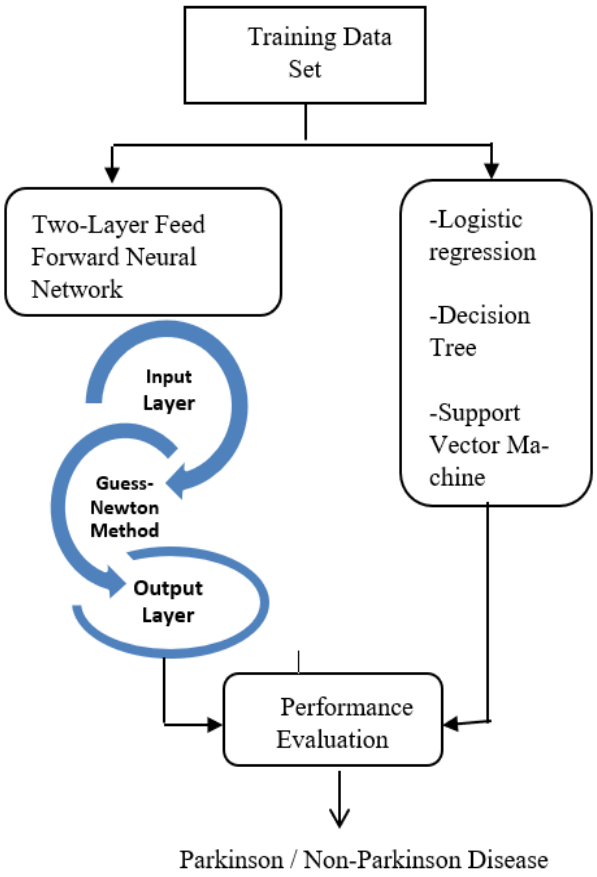

Fig. 3: Flow Diagram of Proposed Work

V. RESULTS AND DISCUSSION

In the work, model is proposed to predict the Parkinson's disease using Parkinson speech dataset. All experiments are performed on apache spark data processing engine for computation in python programming language. In this experiment, 784 instances and 29 attributes were used for determining the accuracy, R-square and f-score of classification models.

Table- III: Performance Measures of Classification Techniques

Classification	Feed-Forward NN using MRGN	Gradient Decent	SVM	LR
Accuracy	**97.98**	93.88	90.17	87.22
F-score	**95**	92	90	86.9
R^2	**0.8**	0.6	0.5	0.4
10-Fold	**96.4**	91.71	88.2	87.5

To classify Parkinson's disease, multi-Layer Feed-Forward neural Network, NN with gradient decent and widely used ML techniques (Logistic Regression and SVM) are performed. SVM performed with RBF kernel.

Table 3 shows the results of proposed experiment in comparison to individual technique. Accuracy of three layers Feed forward neural network with recursive Gauss Newton's method (RGNM) is 94.88% whereas NN with gradient decent is 90.88% and standalone machine learning on whole dataset is: LR; 87.22 and SVM; 89.17% respectively. Results shows that proposed approach outperformed than first order technique and another ML technique.

Fig. 4 shows graphical representation of above performance measure. In this graphs 'X-axis' represents classification techniques and 'Y-axis' is the scale by which performance can measure.

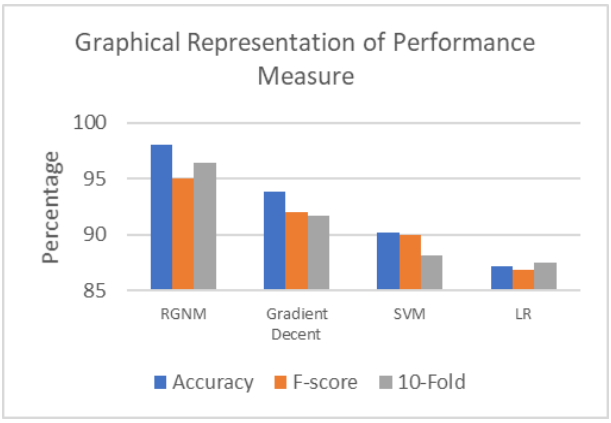

Fig. 4: Graphical representation of performance measure

According to fig. 4, Accuracy of feed-forward neural network with MRGN is better than other classification techniques.

A. Comparative Results of Apache Spark and R-Studio

This experiment is performed on Apache Spark using Python Programming language and R-studio using R programming language. Here Apache spark perform faster in contrast to R because apache Spark is an in-memory data processing engine that contains RDDs (Resilient Distributed Datasets) in their framework

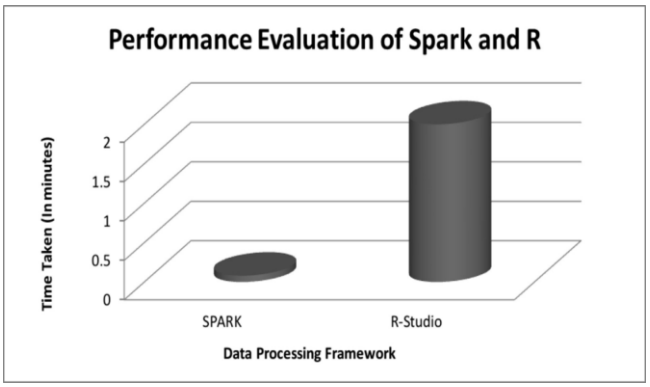

Fig. 5: Performance of SPARK and R-Studio Frameworks

to achieve efficient and faster MapReduce. Basically, resilient distributed datasets are essential unit of data in Apache Spark It helps to perform parallel operation whereas R is an inherited single threaded programming language.

Fig. 5 shows the comparison between performances of both frameworks that a show Apache Spark (5 second or 0.08 minute) is faster than R-studio (2 minutes).

VI. CONCLUSION

Parkinson's disease effects the movements of body. It occurs when certain nerve cells damage in the mind. So, it is required to predict the disease at early stage. In the work, Dataset of Parkinson's speech disease is

extracted using UPDRS rating tool. It contains longitudinal course of Parkinson's disease. In the proposed work, a three-layer feed-forward neural network is constructed using modified recursive gauss-newton's method. Gauss-Newton's method computes curvature information to fetch a more direct path so it is applied to estimate weight for each layer. Proposed work is competitive against state-of-the-art first-order optimization method. The research work is compared with some broadly used standalone machine learning classification techniques such as Support Vector Machine and Logistic Regression to predict the Parkinson's disease. Result shows that proposed technique performs better. According to experimental analysis, Apache Spark is faster as compared to R-Studio framework.

The performance of the results may be improved by increase the number of samples and incorporation of other tests (Gain Ratio test, Chi-square tests and Info Gain test etc.) used in machine learning techniques.

REFERENCES

1. Parkinson's Foundation. (Accessed on 2 Jan 2019). Available: https://www.parkinson.org/Understanding-Parkinsons/Statistics.
2. Hardy, J., Cai, H., Cookson, M.R., Gwinn-Hardy, K. and Singleton, A. (2006). Genetics of Parkinson's disease and parkinsonism. Annals of neurology, 60(4), pp.389-398.
3. Ghassemi, M., Naumann, T., Schulam, P., Beam, A. L., & Ranganath, R. (2018). Opportunities in Machine Learning for Healthcare. arXiv preprint arXiv:1806.00388.
4. Gupta, M., & Gupta, B. (2018, February). A Comparative Study of Breast Cancer Diagnosis Using Supervised Machine Learning Techniques. In 2018 Second International Conference on Computing Methodologies and Communication (ICCMC) (pp. 997-1002). IEEE.
5. Guo, P.F., Bhattacharya, P. and Kharma, N. (2010, June). Advances in detecting Parkinson's disease. In International Conference on Medical Biometrics (pp. 306-314). Springer, Berlin, Heidelberg.
6. Sriram, T.V., Rao, M.V., Narayana, G.S., Kaladhar, D.S.V.G.K. and Vital, T.P.R. (2013). Intelligent Parkinson disease prediction using machine learning algorithms. International Journal of Engineering and Innovative Technology (IJEIT) Volume, 2, pp.44-52.
7. Tsanas, A., Little, M.A., McSharry, P.E., Spielman, J. and Ramig, L.O. (2012). Novel speech signal processing algorithms for high-accuracy classification of Parkinson's disease. IEEE transactions on biomedical engineering, 59(5), pp.1264-1271.
8. Little, M.A., McSharry, P.E., Hunter, E.J., Spielman, J. and Ramig, L.O. (2009). Suitability of dysphonia measurements for telemonitoring of Parkinson's disease. IEEE transactions on biomedical engineering, 56(4), pp.1015-1022.
9. Sapir, S., Ramig, L.O., Spielman, J.L. and Fox, C. (2010). Formant centralization ratio: A proposal for a new acoustic measure of dysarthric speech. Journal of speech, language, and hearing research, 53(1), pp.114-125.
10. Tsanas, A., Little, M.A., McSharry, P.E. and Ramig, L.O. (2011). Nonlinear speech analysis algorithms mapped to a standard metric achieve clinically useful quantification of average Parkinson's disease symptom severity. Journal of the Royal Society Interface, 8(59), pp.842-855.
11. Aich, S., Younga, K., Hui, K. L., Al-Absi, A. A., & Sain, M. (2018, February). A nonlinear decision tree-based classification approach to predict the Parkinson's disease using different feature sets of voice data. In Advanced Communication Technology (ICACT), 2018 20th International Conference on (pp. 638-642). IEEE.
12. Nilashi, M., Ibrahim, O., Ahmadi, H., Shahmoradi, L., & Farahmand, M. (2018). A hybrid intelligent system for the prediction of Parkinson's Disease progression using machine learning techniques. Biocybernetics and Biomedical Engineering, 38(1), 1-15.
13. Ho, A.K., Iansek, R., Marigliani, C., Bradshaw, J.L. and Gates, S. (1999). Speech impairment in a large sample of patients with Parkinson's disease. Behavioural neurology, 11(3), pp.131-137.
14. Harel, B., Cannizzaro, M. and Snyder, P.J. (2004). Variability in fundamental frequency during speech in prodromal and incipient Parkinson's disease: A longitudinal case study. Brain and cognition, 56(1), pp.24-29.
15. Tsanas, A., Little, M. A., McSharry, P. E., & Ramig, L. O. (2010a). Accurate telemonitoring of Parkinson's disease progression by noninvasive speech tests. IEEE transactions on Biomedical Engineering, 57(4), 884-893.
16. Tsanas, A., Little, M.A., McSharry, P.E. and Ramig, L.O. (2010b), March. Enhanced classical dysphonia measures and sparse regression for telemonitoring of Parkinson's disease progression. In Acoustics Speech and Signal Processing (ICASSP), 2010 IEEE International Conference on (pp. 594-597). IEEE.
17. Luchsinger, J.A., Tang, M.X., Stern, Y., Shea, S. and Mayeux, R. (2001). Diabetes mellitus and risk of Alzheimer's disease and dementia with stroke in a multiethnic cohort. American journal of epidemiology, 154(7), pp.635-641.
18. Chen, J., Li, K., Tang, Z., Bilal, K., Yu, S., Weng, C. and Li, K. (2017). A parallel random forest algorithm for big data in a spark cloud computing environment. IEEE Transactions on Parallel & Distributed Systems, (1), pp.1-1.
19. Ozcift, A. (2012). SVM feature selection-based rotation forest ensemble classifiers to improve computer-aided diagnosis of Parkinson disease. Journal of medical systems, 36(4), pp.2141-2147.
20. Das, R. (2010). A comparison of multiple classification methods for diagnosis of Parkinson disease. Expert Systems with Applications, 37(2), pp.1568-1572.
21. Kaladhar, D.S.V.G.K., Nageswara, R.P.V. and Ramesh, N.R.B.L.V. (2010). Confusion matrix analysis for evaluation of speech on Parkinson disease using Weka and MatLab. International Journal of Engineering Science and Technology, 2(7), pp.2734-2737.
22. Anaissi, A., Goyal, M., Catchpoole, D. R., Braytee, A., & Kennedy, P. J. (2016). Ensemble feature learning of genomic data using support vector machine. PloS one, 11(6), e0157330.
23. Hardmeier, C. (2017). Predicting Pronouns with a Convolutional Network and an N-gram Model. In Proceedings of the Third Workshop on Discourse in Machine Translation (pp. 58-62).
24. Güneş, F., Wolfinger, R. and Tan, P.Y. (2017). Stacked Ensemble Models for Improved Prediction Accuracy.
25. Shanthi, S. and Kumar, D. (2012). Prediction of blood glucose concentration ahead of time with feature based neural network. Malaysian Journal of Computer Science, 25(3), pp.136-148.
26. Svozil, D., Kvasnicka, V. and Pospichal, J. (1997). Introduction to multi-layer feed-forward neural networks. Chemometrics and intelligent laboratory systems, 39(1), pp.43-62.
27. Dash, R. (2017). Performance analysis of an evolutionary recurrent Legendre Polynomial Neural Network in application to FOREX prediction. Journal of King Saud University-Computer and Information Sciences.
28. Jabban, T.M., Alali, M.A.E., Mansoor, A.K.Z. and Hamoodi, A.N. (2012). Enhancing the step response curve for rectifier current of HVDC system based on artificial neural network controller. Journal of King Saud University-Engineering Sciences, 24(2), pp.181-192.
29. Ramachandran, P., Zoph, B., & Le, Q. V. (2018). Searching for activation functions.
30. Nasrabadi, N. M. (2007). Pattern recognition and machine learning. Journal of electronic imaging, 16(4), 049901.
31. Clevert, D. A., Unterthiner, T., & Hochreiter, S. (2015). Fast and accurate deep network learning by exponential linear units (elus). arXiv preprint arXiv:1511.07289.
32. Gower, R.M. and Gower, A.L. (2016). Higher-order reverse automatic differentiation with emphasis on the third-order. Mathematical Programming, 155(1-2), pp.81-103.
33. Harrell, F. E. (2015). Ordinal logistic regression. In Regression modeling strategies (pp. 311-325). Springer, Cham.
34. Cox, D. R. (2018). Analysis of binary data. Routledge.
35. Ozcift, A. (2012). SVM feature selection-based rotation forest ensemble classifiers to improve computer-aided diagnosis of Parkinson disease. Journal of medical systems, 36(4), pp.2141-2147.
36. Schlkopf, B., Smola, A. J., & Bach, F. (2018). Learning with kernels: Support vector machines, regularization, optimization, and beyond.
37. Parkinson's Data Set. (Accessed April 2018). Available: https://archive.ics.uci.edu/ml/datasets/Parkinson+Speech+Dataset+with++Multiple+Types+of+Sound+Recordings.
38. Gelman, A., Goodrich, B., Gabry, J., & Vehtari, A. (2018). R-squared for Bayesian regression models. The American Statistician, (just-accepted), 1-6.

39. Galdi, P., & Tagliaferri, R. (2018). Data Mining: Accuracy and Error Measures for Classification and Prediction. Reference Module in Life Sciences, no. January, Elsevier, 1-14.
40. Song, Q., Jiang, H., & Liu, J. (2017). Feature selection based on FDA and F-score for multi-class classification. Expert Systems with Applications, 81, 22-27.
41. Arlot, S., & Celisse, A. (2010). A survey of cross-validation procedures for model selection. Statistics surveys, 4, 40-79.

AUTHORS PROFILE

Ms. Madhuri Gupta is doing PhD from Jaypee Institute of Information technology, Noida, India. Her research area is machine learning, big data analytics and bioinformatics. She is an author of several research papers in reputed journal and conferences. She is professional member of ACM, India.

Dr. Bharat Gupta has completed PhD from University of Westminster, London. He is working as an assistant professor in computer science department. His research area is high performance computing, machine learning, deep learning and data analytics.