International Journal of Research in Advent Technology (IJRAT), Special Issue, March 2019
E-ISSN: 2321-9637
ICCCMIT 2019 organised by M.O.P. Vaishnav College for Women (Autonomous)
Chennai-34, India
Available online at www.ijrat.org

Speedy Convergence of Neural Network Model using Principal Component Analysis for Plant Classification

Vanitha V[1], Sornam M[2*]
Department of Computer Science,Univeristy of Madras, Chennai.[1,2*]
Email: vvani1309@gmail.com[1], *madasamy.sornam@gmail.com*[2*]

Abstract-In the field of Agriculture, recognizing the plant type is getting more focused because of it needs in the medical field to classify the medicinal plant and for the scientist to study the individual properties of the leaf. These problems are generally a multiclass classification in nature. In this proposed work, one of the neural network parameter called learning rate is tuned by gradually reducing it by a small constant value repeatedly once after a certain number of epochs. This diminishing learning rate makes the network to learn better compared to the fixed learning rate. The proposed method has been tested on two different feed-forward neural networks such as Radial basis function neural network and backpropagation neural network with two different classification dataset such as leaf and wheat. To increase the accuracy of the prediction as well as to reduce the computational complexity, dimensionality reduction techniques such as Principal Component Analysis(PCA) and Linear Discriminant Analysis(LDA) has been employed. The result reveals that the PCA based network produces better accuracy when compared to Linear Discriminant Analysis based Network. Compared to raw data and LDA components, the PCA components of the data increases the reliability of the data and it is statistically proven by the metric called R-Square. There is no standard measure to predict the best model, but in this proposed work, Mc Nemar's test is employed to show the significance of the model.

Index Terms-Backpropagation Neural Network; Radial Basis Function Neural Network; Linear Discriminant Analysis; Mc Nemar's Test; Principal Component Analysis; R-Square.

1. INTRODUCTION

Identification of the type of leaf and varieties of wheat is very important in many fields. In our earlier day's prediction based on expert's judgment but sometimes it may lead to a wrong classification and also time consuming task. The misclassification of the plant which is used for medicine extraction may cause death to the patient also. This is because of inter and intra class similarities of the plant. The determination of the type of wheat is very important to fix the market price.

In recent days, the advancement of computer vision technology encourages the botanists and computer scientists to develop the system for automatic identification of the type of plants by using the leaf. Currently, the Artificial Neural Network is employed in a wide range of applications for plant investigation [1]. Eshanirad et al. [2] proposed a neural network model for classifying 32 different types of leaves and obtained an accuracy of 78.46%.The research work of S.Prasad et al. [3], classify 32 different types of plants by using SVM. Meesha et al. [8] employed one of the machine learning algorithmscalled support vector machine for wheat grains classification and obtained an accuracy of 86.8% and 94.5% by using the neural network with Levenberg Marquardt algorithm and the result concluded that neural network is a strong classifier compare to SVM.

Artificial Neural Network is one of the most powerful domains and it is modeled in such a way of how the human brain actually works. Due to the simplicity of ANN, many classification and recognition problem was done by the researchers. Multilayer perceptron is one of the most important artificial neural network model. The combination of Multilayer perceptron [10] along with Backpropagation algorithm gives superior performance and broadly used in different applications where the network training is based on repeatedly updating the weights in between the neurons through the error reduction direction. In this proposed work, the performance of the neural network models analyzed with diminishing learning rate and dimensionality reduction techniques and the result shows that network models give better performance with principal component analysis.

The two different neural network models have been proposed. The first model is the Modified Backpropagation Neural Network with Diminishing Learning Rate(MBP-DLR) and the second model is the Radial Basis Function Neural Network with Diminishing Learning Rate (RBFNN-DLR). The two models were compared to each other by using Mc-Nemar's test to find the best model. The performance was analyzed by using leaf dataset and wheat dataset from the UCI machine learning repository [4]. The rest of the paper organized as follows. In section 2, neural network techniques have been discussed and section 3 explains the dimensionality reduction techniques. Results and Discussions are narrated in section 4, followed by the conclusion in Section 5.

2. NEURAL NETWORK TECHNIQUES

The Neural Network models have the ability to learn the complex and nonlinear relationship between the inputs and outputs which is more important for real world applications.

Two neural networks are proposed namely (i) Modified Backpropagation neural network with diminishing learning rate and (ii) Radial Basis Function Neural Network with diminishing learning rate.

2.1. *Back Propagation Neural Network (BPNN)*

The BPNN is a simple feed forward network which consists of three layers such as input, hidden and the output layer. At the time of training, the input data was fed to the input layer and it is propagated in a forward direction to both the hidden and output layer. This process is called a forward pass. During this time the nodes in the input, hidden and output are calculated and are adjusted the weights between nodes based on an error at the output of the network. The obtained output Y is compared with the desired value to find out the error. Weight between the input to hidden and from hidden to output has been updated based on error helps to minimize error direction and this process is called backward pass.

Proposed Model 1:

Modified Backpropagation Neural Network with Diminishing Learning Rate(MBP-DLR)

In this model, instead of keeping the learning rate as constant, the value diminishing after a fixed number of epochs until it reaches smallest positive value to get a better performance.

Algorithm for MBP-DLR:

(1) Generate random weight matrices A and B between input to hidden layer and hidden to output layer respectively.

(2) For each training pattern from the set T=(X_i,O_i) where X_i is the input vector and O_i is the output vector. Feed the input vector to the input layer.

(3) Compute net input for each hidden node using the following formula

$$u_j = Net_j = \sum_{i=1}^{n} x_i * a_{ij} \ , j = 1,2,..,q \quad (1)$$

(4) Compute output for each hidden nodes

$$g(u_j) = \left(\frac{1}{1+\exp(-u_j)}\right) \quad (2)$$

where g(.) is the sigmoid activation function.

(5) Compute net input for each output node

$$Z_k = Net_k = \sum_{j=1}^{m} u_j * b_{jk} \ , k = 1,2,..,p \quad (3)$$

(6) Calculate output for nodes in the output layer by applying activation function

$$g(Z_k) = \left(\frac{1}{1+\exp(-Z_k)}\right) \quad (4)$$

(7) Calculate gradient of output node

$$Z'_k = (Z_k * (1 - Z_k)) \quad (5)$$

(8) Compute error at output nodes as follows

$$\delta_k = (O_i - Y_i) * Z'_k \quad (6)$$

(9) Error at hidden neuron can be calculated as

$$\delta_j = Zk' * \sum_{k=1}^{m} \delta_k * b_{jk} \quad (7)$$

(10) Proposed diminishing learning rate for output layer weight update:

If (epoch%10==0 and α>0) then

$$B(t+1) = B(t) + ((\alpha - 0.01) * \delta_k * u_j) \quad (8)$$

Else

$$B(t+1) = B(t) + (\alpha * \delta_k * u_j) \quad (9)$$

where α is the learning rate

(11) Hidden layer weight update using proposed diminishing learning rate method:

If (epoch%10==0 and α>0) then

$$A(t+1) = A(t) + ((\alpha - 0.01) * \delta_j * X_i) \quad (10)$$

Else

$$A(t+1) = A(t) + (\alpha * \delta_j * X_i) \quad (11)$$

(12) Repeat Step 2 to Step 11 for 1000 times

2.2 Radial Basis Function Neural Network (RBFNN)

RBFNN is a feed forward neural network consists of input, hidden and output layer and it can be applied for function approximation, classification, time series prediction problems and so on [9]. The training of RBFNN network is mainly based on various parameters such as center, widths and weight connection between hidden to output neurons. The center can be calculated by using various unsupervised clustering algorithms. The activation function for the hidden neuron is usually a Gaussian function. Compare to Batch training of RBFNN,the online training giving better accuracy [10]. In this work, K-

Means Clustering algorithm used for predicting centers**.**

The following is an algorithm for K-Means Clustering.

(1) Place K different points in the input object space that are being clustered to represent an initial group of the centroid.
(2) Form K different clusters by assigning objects to the group which is closest to the centroid.
(3) Once grouping has been done, recalculate the position of the K centroid.
(4) Repeat steps 2 and 3 until the position of centroid cannot be moved further.

Proposed Model 2:

Radial Basis Function Neural Network with Diminishing Learning Rate (RBFNN-DLR)

In RBFNN online training algorithm, one of the important hyper-parameter is the learning rate. Instead of keeping it as constant, the learning rate was getting diminishing gradually which makes the network to learn faster.

Algorithm for RBFNN-DLR

(1) Compute center using K means clustering algorithm
(2) Calculate sigma (spread radius).
(3) Generate random weight matrix V between hidden to output layer
(4) Pass input to the network
(5) Calculate output for hidden neurons using Gaussian activation function

$$H_j=\exp\left(\frac{-(x_k-c_j)^2}{2\sigma^2}\right) \quad (12)$$

Where H_j is hidden neuron, C_j is center and σ is spread.

(6) Compute net input for each output node

$$Z_k = Net_k = \sum_{j=1}^{m} h_j * v_{jk}\ , k = 1,2,..,p \quad (13)$$

(7) Calculate output for nodes in the output layer by applying activation function

$$g(Z_k) = \left(\frac{1}{1+\exp(-Z_k)}\right) \quad (14)$$

where g(.) is the sigmoid activation function

(8) Calculate slope for an output value

$$Z'_k = (Z_k * (1 - Z_k)) \quad (15)$$

(9) Compute error for output nodes as follows

$$\delta_k = (O_i - Y_i) * Z'_k \quad (16)$$

(10) Proposed diminishing learning rate for output layer weight update:

If (epoch%10==0 and α>0) then

$$V(t+1) = V(t) + ((\alpha - 0.01) * \delta_k * X_i) \quad (17)$$

Else

$$V(t+1) = V(t) + (\alpha * \delta_k * X_i) \quad (18)$$

where α is the learning rate

(11) Repeat Step 2 to Step 8 for 1000 times

3. DIMENSIONALITY REDUCTION TECHNIQUES

Dimensionality reduction is one of the feature engineering technique. Compare to other dimensionality reduction techniques, in PCA the information loss is minimal. The familiar supervised dimensionality reduction technique is a Linear discriminant analysis which works well if the data is normally distributed.

3.1. *Principal Component Analysis*

The most powerful unsupervised dimensionality reduction technique is Principal Component Analysis. It is used to transform the dependent correlated variables into independent uncorrelated variables and these should be termed as Principal Factors.

The procedure for PCA's as follows

(1) PCA capture the maximum variance of all the predictors and make it as the first factor.
(2) Remove the variance explained by the first factor from the dataset and find the next maximum variance from the second factor.
(3) PCA returns n factors if we have n dimension in the dataset.
(4) Each and every factor are called Principal Components.

3.2. *Linear Discriminant Analysis*

LDA is a supervised dimensionality reduction technique which is widely used in statistics. LDA work by mapping data from higher dimensional space into lower dimensional space.

The three steps involved in LDA are

(1) Calculate the distance between the means of different classes and it is called between-class variance.
(2) Compute within-class variance.
(3) Map the data into lower dimensional space by increasing between-class variance and decreasing within-class variance.

4. RESULTS AND DISCUSSIONS

4.1 Dataset Description

For experimentation, two datasets have been taken from UCI Machine learning repository for classifying

32 different types of Leaf [13] and 3 different type of Wheat [12].

4.1.1 Leaf Dataset

The leaf dataset contains 340 instances for 32 different types of leaf. The shape attributes are numbered from 3 to 8 and attributes from 9 to 15 contains the textual properties of the leaf.

Table 1.Shape features of leaf

S. No	Shape Feature	Description
1	Eccentricity	Eccentricity of the ellipse with identical second moments to I. This value ranges from 0 to 1
2	Aspect Ratio	The aspect ratio is defined as the quotient D(I)/D values close to 0 indicate an elongated shape
3	Elongation	It is the ratio between the diameter of the largest inscribed circle and the diameter of the smallest circumscribed circle
4	Solidity	It is the measures of how well Eccentricity fits a convex shape
5	Stochastic Convexity	This variable extends the usual notion of convexity of topological sense, using sampling to perform the calculation
6	Isoperimetric Factor	Curvy Intertwined contours
7	Maximal Indentation Depth	It is the maximal indentation depth of the function
8	Lobedness	This feature characterizes how lobed a leaf is

Table 2. Textural features of leaf

S.No	Textural Feature	Description
9	Average Intensity	Average Intensity is defined as the mean of the intensity image, m.
10	Average Contrast	Average contrast is the standard deviation of the intensity image
11	Smoothness	It measures the relative smoothness of the intensities in a given region.
12	Solidity	It is the measures of how well Eccentricity fits a convex shape
13	Third Moment	It is a measure of the intensity histogram's skewness
14	Uniformity	It reaches maximum value when all intensity levels are equal
15	Entropy	A measure of intensity randomness

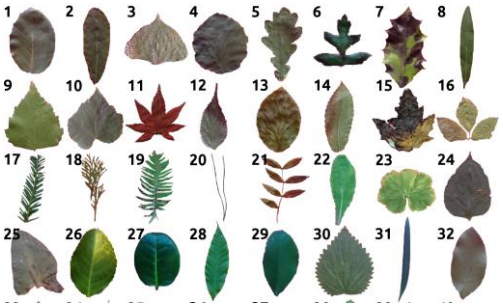

Fig. 1. Leaf Images

4.1.2Wheat Dataset

There are 210 records in the wheat dataset. The seven geometrical parameters are used to measure the kernels of wheat to classify three different type of wheats such as Kama, Rosa and Canadian. The parameters for predictions are,

(1) area A
(2) perimeter P
(3) compactness $C = 4*pi*A/P^2$
(4) length of kernel
(5) width of kernel
(6) asymmetry coefficient
(7) length of kernel groove.

4.2 Dimensionality Reduction Result

4.2.1 PCA Result of Leaf Dataset

By applying PCA in the leaf dataset, the dimension was drastically reduced from 15 attributes to 7 principal components which covers the maximum variance of 95%.

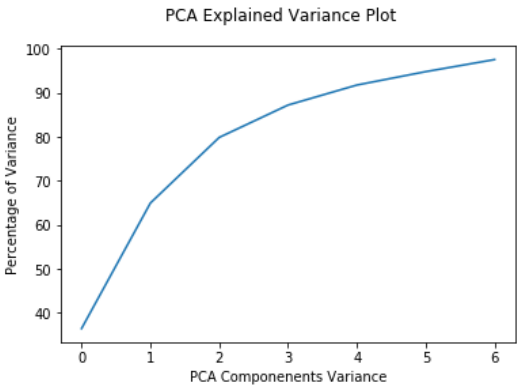

Fig. 2. PCA explained variance plot for leaf dataset

4.2.2 PCA Result of Wheat Dataset

The PCA obtained the 4 components from 7 attributes which covers 95% variance of the dataset.

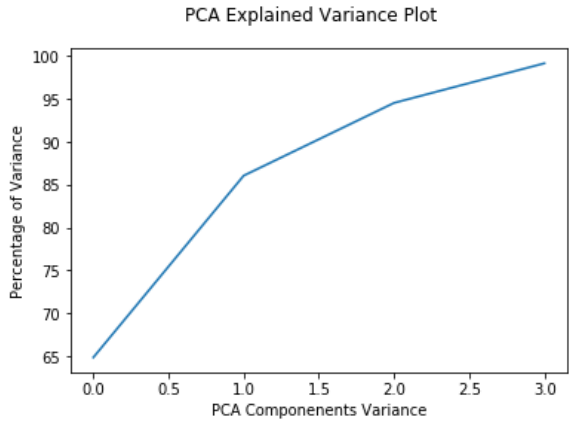

Fig. 3. PCA explained variance plot for Wheat dataset

4.2.3LDAResult of wheat Dataset

The LDA returns 4 Components from 7 predictors which covers 100% of variance of the dataset.

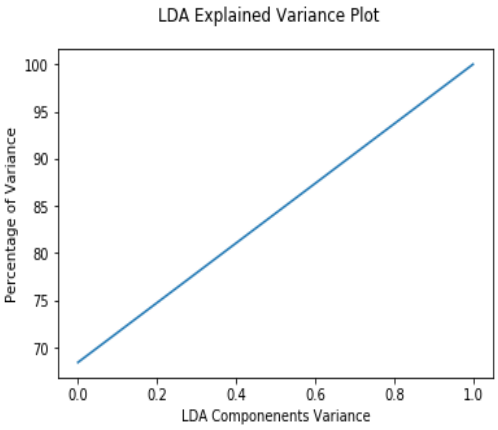

Fig. 4. LDA explained variance plot for Wheat dataset

4.2.4LDAResult of Leaf Dataset

The LDA returns only 5 components from 15 attributes which capture 90% of variance of the dataset.

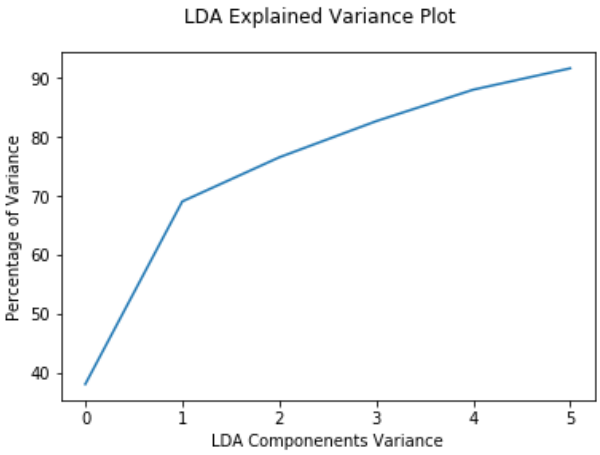

Fig. .5. LDA explained variance plot for Leaf dataset

4.3 Neural Network Results

The dataset was divided into 70% for training and 30% for testing. The stopping criteria is a fixed number of 1000 epochs.

Table III and Table IV are used to depicts the results obtained from the two different proposed methods specifically to differentiate the accuracy of optimized learning rate and fixed learning rate. Compared to fixed learning rate, the performance better in the case of reducing learning rate.

From the leaf dataset, 238 records were used for training and 102 records for testing. In wheat dataset, 147 records were used for training and 63 records for testing.

Table 3. Comparative performance results of proposed mbp-ldr with standard bpn

Methodologies	**Leaf Dataset**		**Wheat Dataset**	
	Training	*Testing*	*Training*	*Testing*
MBP-DLR with Raw Data	98%	97%	97.8%	97.1%
BPN with Raw Data	95%	92%	93.81%	91.8%
MBP-DLR with LDA	97%	94%	96.5%	94%

Methodologies	**Leaf Dataset**		**Wheat Dataset**	
	Training	*Testing*	*Training*	*Testing*
BPN with LDA	91%	88%	92.5%	91%
MBP-DLR with PCA	99.63%	99.46%	100%	100%
BPN with PCA	95%	94.46%	99%	98%

Table 4. Comparative performance results of proposed RBFNN-ldr with standard RBFNN

Methodologies	**Leaf Dataset**		**Wheat Dataset**	
	Training	*Testing*	*Training*	*Testing*
RBFNN -DLR with Raw Data	70%	73%	92%	90.76%
RBFNN with Raw Data	66%	64%	89.7%	88%
RBFNN -DLR with LDA	60%	59%	78%	77.25%
RBFNN with LDA	58%	60%	70.34%	69%
RBFNN -DLR with PCA	82%	80%	96%	97%
RBFNN with PCA	80.3%	79.87%	94.8%	96%

4.4Reliability of Data

The R-Square is the statistical metric used to predict the reliability of the data. It is used to measure how the data is closed to fit the regression line. It is also known as the coefficient of determination. In the examination dataset, R-Squared value estimated for the real value and for the principal component of the dataset. The formula to compute the r-square value for the multiple predictors as follows.

The formula for fitted value is

$$Y' = B_0 + B_1X_1 + B_2X_2 + B_3X_3 + \cdots + B_pX_p \quad (19)$$

where Y' is the expected output variable, X_1 through X_p are independent variables and from B_1 to B_p are estimated regression coefficients.
The overall measure of the goodness of fit is

$$SSE = \sum_{i=1}^{n}(Y_i - Y')^2 \quad (20)$$

where SSE is sum of square error which is used to identify how much the data point differ from the estimated regression line, Yi is the data points and Y' is the regression line.

The total sum of squares

$$SSTO = \sum(Y_i - Y)^2 \quad (21)$$

The R-Square value computed using the formula

$$R - Square = 1 - \frac{SSE}{SSTO} \quad (22)$$

SSE – Sum of Squares of Error for Regression Line
SST – Total sum of Squares of Error

Table 5. R-Square value for dataset

	Wheat Dataset	**Leaf Dataset**
Raw Data	0.751	0.747
PCA Components	0.998	0.787
LDA Components	0.765	0.648

The result shows that the PCA components of the dataset gives more reliable value for prediction than for raw data and LDA components of the dataset.

4.4Mc Nemar's Test

In recent days, different models have been proposed so there is a need of standard metric to find how the model is significant. Mc Nemar's test is an effective way to find the best model and it works by comparing two different models on a single dataset. From the results given in Table III and IV, the performance of proposed MBP-DLR was superior when it is

compared with RBFNN-DLR. This can be shown by applying Mc-Nemar's test to compare two different classifiers to find the best. The following steps used to conduct the Mc Namer's Test.

(1) Compute the contingency table.
(2) Calculate statistics by using the following formula

$$Statistics = \frac{(Y/N - N/Y)^2}{Y/N + N/Y)} \quad (23)$$

(3) The Null hypothesis means that the two cases disagrees to the same amount
(4) Fix the Alpha value as 0.05. For the selection of significance level, calculate the p-value and it is interpreted as follows

(5) If P>alpha, then fail to reject null hypothesis there is no difference in the disagreement
(6) if P<alpha, then there is a significant difference in the disagreement.

Table 6: Structure of Contingency table

	Classifier2 Correct	**Classifier2 Incorrect**
Classifier1 Correct	Y/Y	Y/N
Classifier1 Incorrect	N/Y	N/N

4.4.1 Mc Nemar's test for Wheat Dataset

The Mc Nemar's test is applied to compare the two different proposed models with PCA components. The result shows that there is a significant difference and BPN with PCA is the best Classifier when it is compared with RBFNN with PCA.

Table 7. Contingency table for wheat dataset

	RBFNN-DLR with PCA (correct)	**RBFNN-DLR with PCA (incorrect)**
MBP-DLR with PCA(correct)	191	19
MBP-DLR with PCA(incorrect)	0	0

statistic=0.000, p-value=0.000

Since the P value is less than alpha value, hence there is a significant difference between the models.

4.4.2Mc Nemar's test for Leaf Dataset

The result of Mc Nemar's shows that there is a significant difference between the two proposed models.

Table 8. Contigency table for leaf dataset

	RBFNN-DLR with PCA(correct)	**RBFNN-DLR with PCA(incorrect)**
MBP-DLR with PCA(correct)	272	64
MBP-DLR with PCA(incorrect)	1	3

Statistic=1.000, p-value=0.000

Since the P value is less than alpha value, it can be inferred that there is a significant difference between the models.

5. CONCLUSION

In this proposed work, the focus is on optimizing the learning rate parameter of the neural network which works effectively to improve the accuracy of the network. In general, the purpose of the learning rate is to make the network to learn faster, but the presence of same value of learning rate initially speed up the learning later on the network having a problem called overfitting. To overcome this, the value of the learning rate reduced by a small constant value after some fixed number of epochs until it becomes a small positive value, which helps to increase the accuracy of prediction. To show the significance of the model for classification, Mc-Nemar's test was employed. Dimensionality reduction is also more useful to improve the performance as well as speed up the network because of reducing the number of attributes for prediction. Compared to LDA, the PCA works well for the experimental dataset namely leaf and wheat.

REFERENCES

[1]. Baldi, Ada, Camilla Pandolfi, Stefano Mancuso, and Anna Lenzi. "A leaf-based back propagation neural network for oleander (Nerium oleander L.) *cultivar identification." Computers and Electronics in Agriculture 142 (2017): 515-520.*
[2]. Ehsanirad, Abdolvahab, and Y. H. Sharath Kumar. "Leaf recognition for plant classification using

GLCM and PCA methods." *Oriental Journal of Computer Science and Technology* 3.1 (2010): 31-36.
[3]. Prasad, Shitala, Krishna Mohan Kudiri, and R. C. Tripathi. "Relative sub-image based features for leaf recognition using support vector machine." *Proceedings of the 2011 International Conference on Communication, Computing & Security*. ACM, 2011.
[4] Asuncion, Arthur, and David Newman. "UCI machine learning repository." (2007).
[5] Zhou, Zhi-Hua, and Yuan Jiang. "Medical diagnosis with C4. 5 rule preceded by artificial neural network ensemble." *IEEE Transactions on information Technology in Biomedicine* 7.1 (2003): 37-42.
[6] Gang, Ling, Brijesh Verma, and S. Kulkami. "Experimental analysis of neural network based feature extractors for cursive handwriting recognition." *Neural Networks, 2002. IJCNN'02. Proceedings of the 2002 International Joint Conference on*. Vol. 3. IEEE, 2002.
[7]. Charalambous, Chris, Andreas Charitou, and Froso Kaourou. "Comparative analysis of artificial neural network models: Application in bankruptcy prediction." *Annals of operations research* 99.1-4 (2000): 403-425.
[8]. Punn, Meesha, and Nidhi Bhalla. "Classification of wheat grains using machine algorithms." International Journal of Science and Research (IJSR) 2.8 (2013): 363-366.
[9]. Antolìn, Ascensiòn Gallardo, Juan Pascual Garcìa, and Josè Luis Sancho Gòmez. "Radial Basis Function Networks and their Application in Communication Systems." Computational Intelligence. Springer, Boston, MA, 2007. 109-130.
[10]. Karayiannis, Nicolaos B. "Gradient descent learning of radial basis neural networks." Neural Networks, 1997., International Conference on. Vol. 3. IEEE, 1997.
[11]. khalil Alsmadi, Mutasem, Khairuddin Bin Omar, Shahrul Azman Noah, and Ibrahim Almarashdah. "Performance comparison of multi-layer perceptron (Back Propagation, Delta Rule and Perceptron) algorithms in neural networks." Advance Computing Conference, 2009. IACC 2009. IEEE International. IEEE, 2009.
[12]. https://archive.ics.uci.edu/ml/datasets/seeds
[13]. https://archive.ics.uci.edu/ml/datasets/leaf