International Journal of Recent Technology and Engineering (IJRTE)
ISSN: 2277-3878 (Online), Volume-8 Issue-2, July 2019

Predicting Forest Fires using Supervised and Ensemble Machine Learning Algorithms

R. Rishickesh, A. Shahina, A. Nayeemulla Khan

Abstract*: Forest fires have become one of the most frequently occurring disasters in recent years. The effects of forest fires have a lasting impact on the environment as it lead to deforestation and global warming, which is also one of its major cause of occurrence. Forest fires are dealt by collecting the satellite images of forest and if there is any emergency caused by the fires then the authorities are notified to mitigate its effects. By the time the authorities get to know about it, the fires would have already caused a lot of damage. Data mining and machine learning techniques can provide an efficient prevention approach where data associated with forests can be used for predicting the eventuality of forest fires. This paper uses the dataset present in the UCI machine learning repository which consists of physical factors and climatic conditions of the Montesinho park situated in Portugal. Various algorithms like Logistic regression, Support Vector Machine, Random forest, K-Nearest neighbors in addition to Bagging and Boosting predictors are used, both with and without Principal Component Analysis (PCA). Among the models in which PCA was applied, Logistic Regression gave the highest F-1 score of 68.26 and among the models where PCA was absent, Gradient boosting gave the highest score of 68.36.

***Index Terms*: Forest Fires, Principal Component Analysis, Supervised Learning Algorithms, Ensemble Learning Algorithms.**

I. INTRODUCTION

Forest fires (also known as wildfires) have become one of the most frequently occurring disasters in recent times. Large acres of forest area are getting destroyed due to these wildfires. One of the main reasons behind the occurrence of forest fires is global warming which attributes to the increase in average temperature of the earth. The other reasons are due to lightning, during thunderstorms, and human negligence. Each year an average of 1.2 million acres of forest in the US get destroyed due to the wildfires. In India forest fires have increased by 125% between the years 2016 and 2018.

Forest fires can lead to deforestation which has a lot of negative impact on the human society. The usual process of dealing with wildfires is where the satellite images of the forest fires are captured and the authorities are notified by its occurrence and the measures are taken to stop it [1]. But the above process will happen only after the occurrence of the wildfires and before even attempting to take the first step of action there will already be few acres of forest area destroyed. The whole process should be known in advance and should be mitigated. The process should not only be time efficient but also be cost efficient.

Data mining is one such efficient approach in which the forest fires can be predicted based on their past occurrences [2], [3], [4]. Data mining requires an authentic and a clean set of data for prediction. If the dataset is not clean or if there are many unknown values then those values must be taken care of before we use them for modeling. The dataset present in the UCI Machine learning repository about the forest fires is used for prediction [5]. The tabulated data is about the wildfires that happened in the Montesinho park situated in Portugal.

Cortez et al. proposed a related work to predict the area burned by the forest fires using the dataset [5]. Initially, the feature 'area' was transformed using $ln(x+1)$ function. Data mining models were applied and fitted. Post-processing was done on the outputs with the inverse of $ln(x+1)$ transform. The experiment was conducted using 10-fold (cross-validation) x 30 runs. The metrics used there for regression were MAD (Mean Absolute Deviation) and RMSE (Root Mean Square Error). Support Vector Machines with Gaussian kernel using 4 features, namely temperature, relative humidity, wind speed and rain, and Naïve mean predictor obtained the best MAD and RMSE values, respectively. The results also suggest that SVM predicted small fires with better accuracy. G E Sakr et al. proposed a forest fire prediction method based on meteorological data [6]. The results suggest that SVM gave a higher accuracy for a two-class prediction and for a four-class prediction.

In this paper instead of predicting the area of the forest burned, the occurrence of the forest fire is predicted using classification techniques. Apart from the learning algorithms like Logistic regression, Support Vector Machine and Random Forest, this paper also takes into account the ensemble Bagging and Boosting classifiers [7], [8], [9]. The performance metric used in this work is the F-1 score which considers both precision and recall as an evaluating measure [10], [11].

II. EXPLORATORY DATA ANALYSIS

The dataset consists of the following features: X and Y axes special coordinates within the park, Fine Fuel Moisture Code (FFMC), Duff Moisture Code (DMC), Drought Code (DC) and Initial Spread Index (ISI). The other features used are temperature, relative humidity, wind speed, outside rain and the forest area that is burnt. All these features have been collected on all days for an entire year from January to December.

Revised Manuscript Received on 30 July 2019.
***** Correspondence Author
R. Rishickesh*, Department of Information Technology, SSN College of Engineering, Kalavakkam-603110, India
A. Shahina, Department of Information Technology, SSN College of Engineering, Kalavakkam-603110, India .
A. Nayeemulla Khan, School of Computing Science and Engineering, VIT University, Chennai-600127, India

	X	Y	month	day	FFMC	DMC	DC	ISI	temp	RH	wind	rain	area
0	7	5	mar	fri	86.2	26.2	94.3	5.1	8.2	51	6.7	0.0	0.0
1	7	4	oct	tue	90.6	35.4	669.1	6.7	18.0	33	0.9	0.0	0.0
2	7	4	oct	sat	90.6	43.7	686.9	6.7	14.6	33	1.3	0.0	0.0
3	8	6	mar	fri	91.7	33.3	77.5	9.0	8.3	97	4.0	0.2	0.0
4	8	6	mar	sun	89.3	51.3	102.2	9.6	11.4	99	1.8	0.0	0.0
5	8	6	aug	sun	92.3	85.3	488.0	14.7	22.2	29	5.4	0.0	0.0
6	8	6	aug	mon	92.3	88.9	495.6	8.5	24.1	27	3.1	0.0	0.0
7	8	6	aug	mon	91.5	145.4	608.2	10.7	8.0	86	2.2	0.0	0.0
8	8	6	sep	tue	91.0	129.5	692.6	7.0	13.1	63	5.4	0.0	0.0
9	7	5	sep	sat	92.5	88.0	698.6	7.1	22.8	40	4.0	0.0	0.0

Fig.1 Sample 10 rows of the dataset.

Fig.1 represents a sample (first 10 rows) of the UCI forest fire dataset. The feature 'month' and 'day' are categorical features whereas the rest of them are continuous features.

	X	Y	month	day	FFMC	DMC	DC	ISI	temp	RH	wind	rain	area	fire
0	7	5	mar	fri	86.2	26.2	94.3	5.1	8.2	51	6.7	0.0	0.0	0.0
1	7	4	oct	tue	90.6	35.4	669.1	6.7	18.0	33	0.9	0.0	0.0	0.0
2	7	4	oct	sat	90.6	43.7	686.9	6.7	14.6	33	1.3	0.0	0.0	0.0
3	8	6	mar	fri	91.7	33.3	77.5	9.0	8.3	97	4.0	0.2	0.0	0.0
4	8	6	mar	sun	89.3	51.3	102.2	9.6	11.4	99	1.8	0.0	0.0	0.0

Fig.2 The dataset with a newly added column 'fire'.

As given in the Fig.2, the 'fire' feature is derived from the 'area' feature where the values of 0 in the column 'area' corresponds to 0 in column 'fire' and values above 0 corresponds to 1 in the column 'fire', denoting the presence of forest fires.

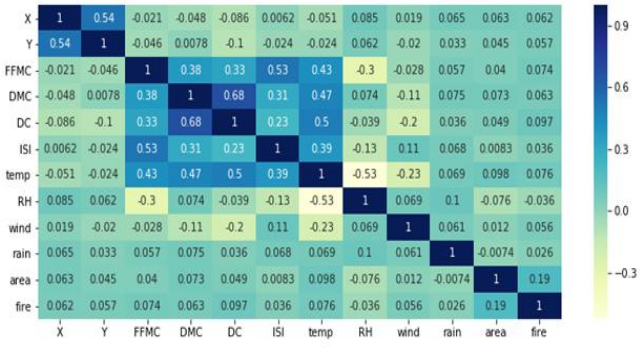

Fig.3 The correlation plot between different features in the dataset.

The dataset is checked for null values in any of the columns and cleaned. The column 'fire' is created using the column area where the value of 'fire' is 1 if the value of 'area' is greater than 0 and is 0 if the value of 'area' is also equal to 0.

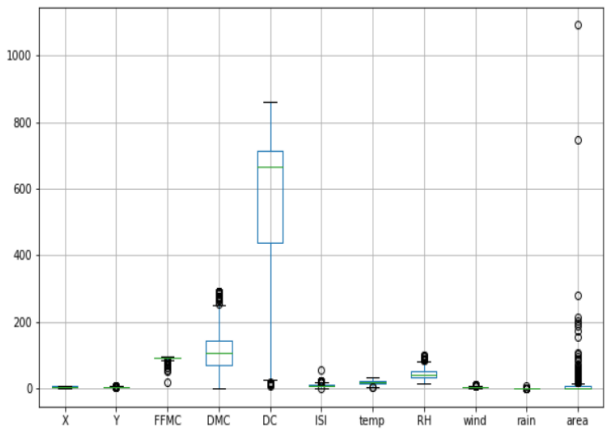

Fig.4 Box plot showing the variation in values for the different features of the dataset.

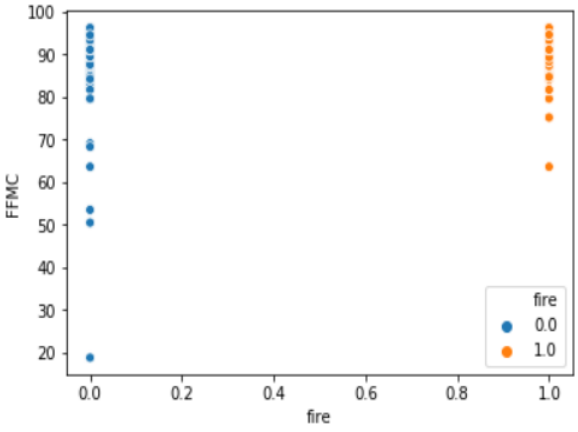

Fig.5 Plot consisting of variation of FFMC with respect to 'fire' and is hued with respect to 'fire'. The value of 0 for 'fire' means there is no occurrence of forest fires and 1 represents the occurrence of forest fires.

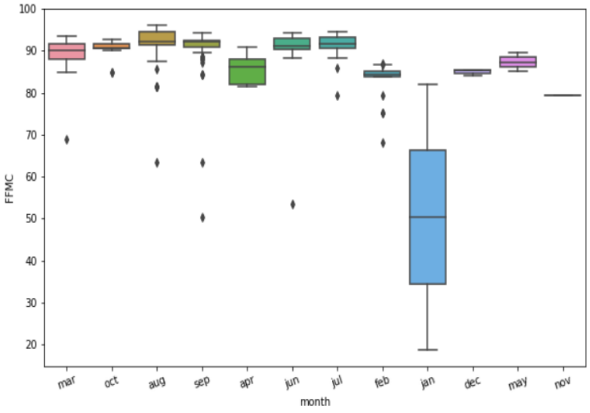

Fig.6 Box plot representation of the variance of FFMC with respect to each month.

The values of FFMC in the dataset are constantly and continuously spread with many values slightly deviating away from the mean values. The range of values is very less as shown in Fig.4. The value of FFMC varies a lot in the month of January.

Fig.6 suggests that the range of values for FFMC in the month of January is closed and does not have any outliers. The values remain constant in the month of November. It is strongly correlated with the values of ISI with a positive coefficient of 0.53 as shown in the Fig.3. It shows a considerable amount of negative correlation with RH at -0.3. Fig.5 suggests that the values of FFMC when there are no forest fires, predominantly take values above 80 and very few lie below 70. When there is forest fires the values of FFMC are always above 60.

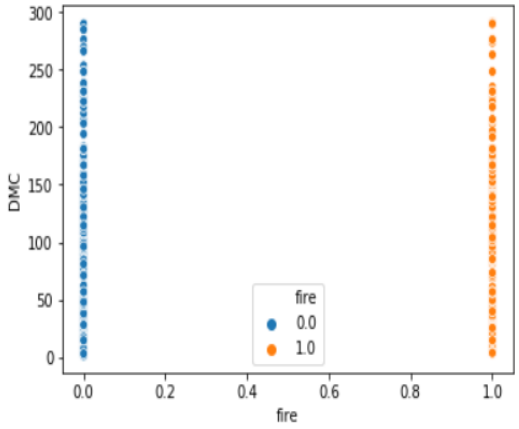

Fig.7 Plot consisting of variation of DMC with respect to 'fire' and is hued with respect to 'fire'. The value of 0 for 'fire' means there is no occurrence of forest fires and 1 represents the occurrence of forest fires.

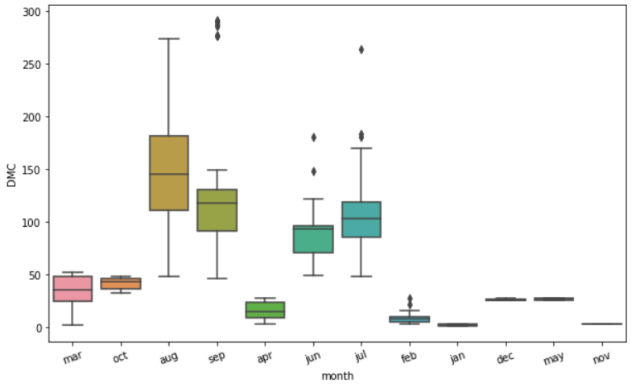

Fig.8 Box plot representation of the variance of DMC with respect to each month.

The values of DMC are continuously spread and the values have a larger range. There are quite a few values that deviate from the mean values as shown in the Fig.4. Fig.8 suggests that the variation in the values is significant at times (e.g., August), while at other times it is very minimal (e.g., November). Sometimes, the values tend to deviate significantly from the mean value (e.g., July and September). The values are strongly correlated with the values of DC where the correlation coefficient's value is 0.68 as shown in the Fig.3. It is negatively correlated only with the features 'X' and 'wind'. Fig.7 illustrates that the values of DMC are equally spread right frosm 0 to 300 when there is no forest fire. Even when there are no forest fires, the values are equally spread from 0 to 300.

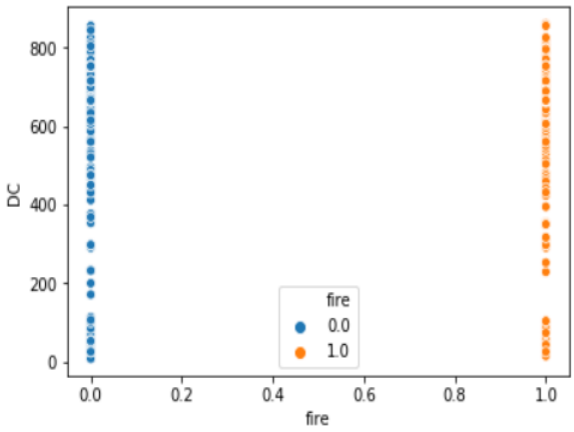

Fig.9 Plot consisting of variation of DC with respect to 'fire' and is hued with respect to 'fire'. The value of 0 for 'fire' means there is no occurrence of forest fires and 1 represents the occurrence of forest fires.

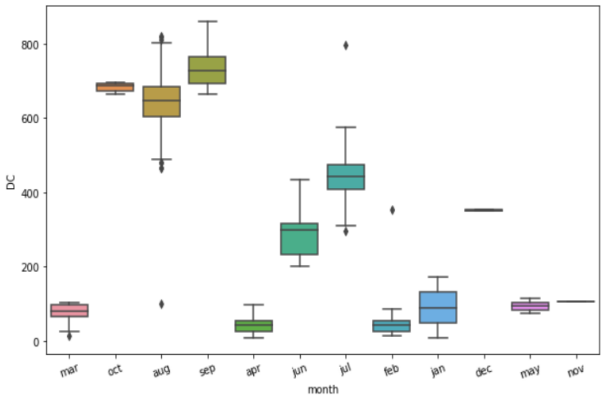

Fig.10 Box plot representation of the variance of DC with respect to each month.

The values of DC have the largest value range among all the features in the dataset (see Fig.4). They show variation in such a way that the maximum value found is just above 800 and minimum value is 0. With respect to each month as illustrated in the Fig.10, the values of DC do not vary a lot having very small ranges. Even though the ranges are small, months like January, June, August depict wide range of values. Fig.3 suggests DC has a strong correlation with DMC and also a positive correlation value of 0.5 with temperature. They do not show a strong negative correlation with any other features. Similar to DMC (see Fig.9), the values of DC are spread equally in both the cases.

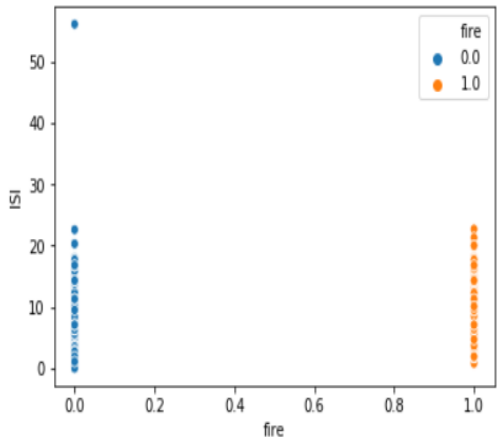

Fig.11 Plot consisting of variation of ISI with respect to 'fire' and is hued with respect to 'fire'.

The value of 0 for 'fire' means there is no occurrence of forest fires and 1 represents the occurrence of forest fires.

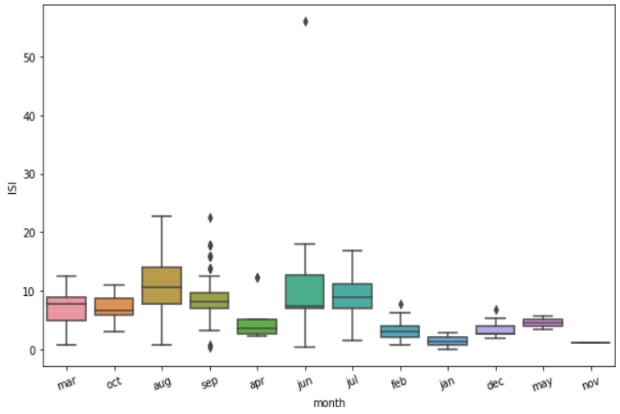

Fig.12 Box plot representation of the variance of ISI with respect to each month.

The range of values of ISI is small and most of the values are very close to each other (see Fig.4). There are few values that lie outside the range. The values vary a lot in the month of August, June, July as illustrated in the Fig.12. They do not show variations in the month of November. ISI values show a good correlation with the values of temperature. They do not show any strong negative correlation as Fig.3 suggests. The values of ISI when there are no forest fires, predominantly take values below 30 and just one lie above 30. When there is a forest fire, the values of ISI are always below 30 as shown in the Fig.11.

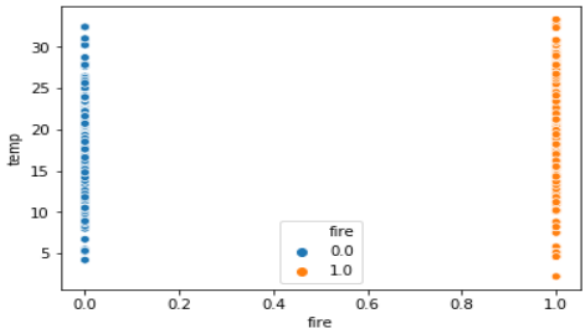

Fig.13 Plot consisting of variation of temp with respect to 'fire' and is hued with respect to 'fire'. The value of 0 for 'fire' means there is no occurrence of forest fires and 1 represents the occurrence of forest fires.

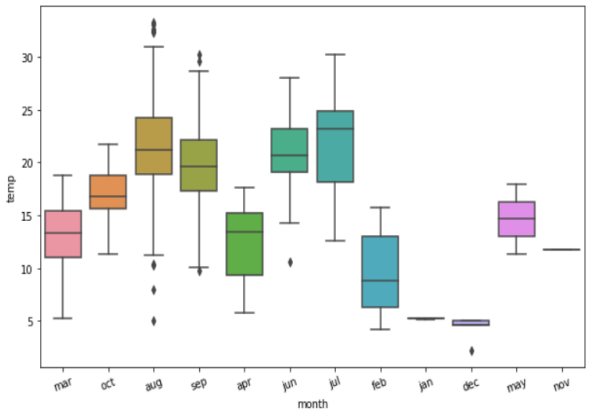

Fig.14 Box plot representation of the variance of 'temp' with respect to each month.

The values of temperature are within a small range and are very close to each other as shown in the Fig.4. They are measured in terms of degree Celsius. The values that deviate from mean are very less. Fig.14 suggests that they show great variation in the month of February, April and July. The variations are very less in the month November, December and January. The positive correlations of the temperature feature with other features are very weak (correlation coefficient less than 0.5). They show a considerable amount of negative correlation with the feature RH (see Fig.3). The values of temperature are spread equally when there is presence of forest fires and also during absence of forest fires (See Fig.13).

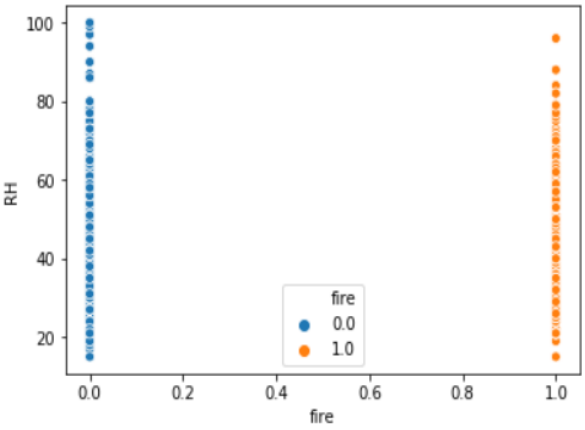

Fig.**15 Plot consisting of variation of RH with respect to 'fire' and is hued with respect to 'fire'. The value of 0 for 'fire'** means there is no occurrence of forest fires and 1 represents the occurrence of forest fires.

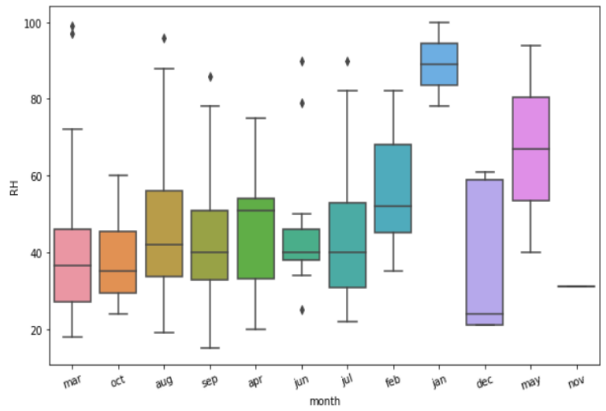

Fig.16 Box plot representation of the variance of RH with respect to each month.

The values of Relative humidity (RH) are within a small range and are very close to each other (see Fig.4). The number of values that deviate from the mean are very less. The values vary a lot in the month of December and May without any outlying value as shown in the Fig.16. Like the other features, RH values do not show any variations in the month of November. They are weakly correlated with all the features except temperature as illustrated in the Fig.3. Fig. 15 suggests that the values of RH are spread equally when there is presence of forest fires and also during absence of forest fires.

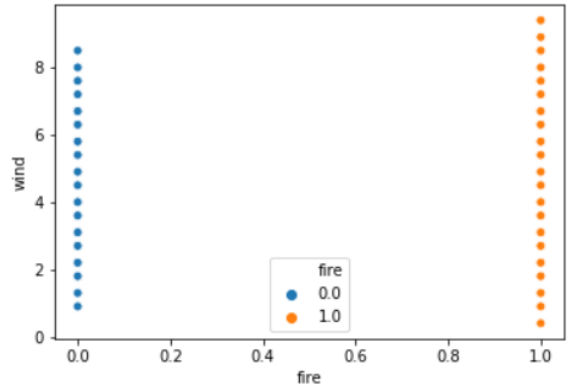

Fig.17 Plot consisting of variation of 'wind' with respect to 'fire' and is hued with respect to 'fire'. The value of 0 for 'fire' means there is no occurrence of forest fires and 1 represents the occurrence of forest fires.

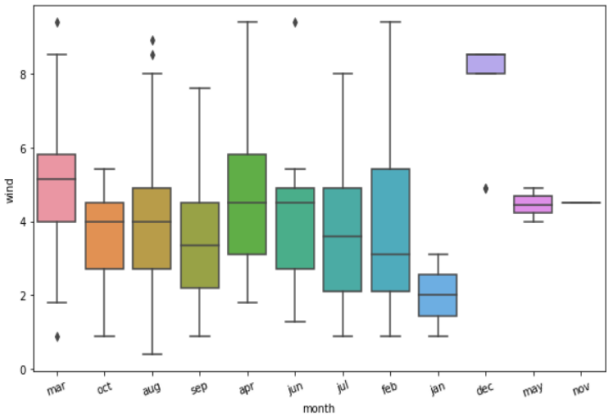

Fig.18 Box plot representation of the variance of 'wind' with respect to each month.

The values of wind speed are very less and are closer to each other (See Fig.4). The outlier count is very less. The values vary a lot in the month of February and July. They do not vary in the month of November as shown in Fig.18 and are weakly correlated with all the features. The speed of the wind is measured in terms of Km/hr and range from 0.4 to 9.4 Km/hr. The values of wind are spread equally when there is presence of forest fires and also during absence of forest fires with the range being slightly larger during the absence of forest fires as shown in the Fig.17.

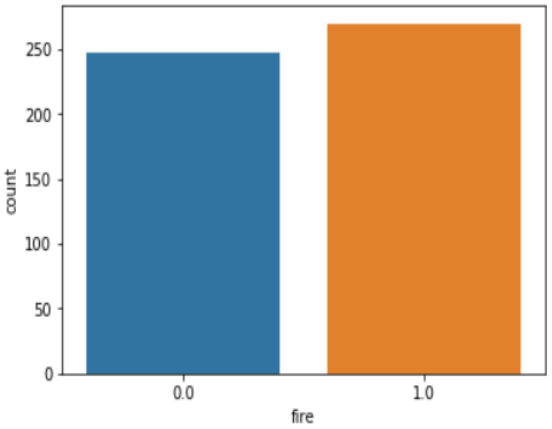

Fig.19 Bar graph showing the frequency of forest fires with respect to its occurrence.

From the dataset it is inferred that the number of times that the wildfires have happened is 270 and number of times the wildfires have not happened is 247 as shown in the Fig.19.The values of FFMC when the wildfires took place have a larger range than when the wildfires did not happen. The values of DC, DMC, temperature, RH and wind speed have a bigger range for both the cases.

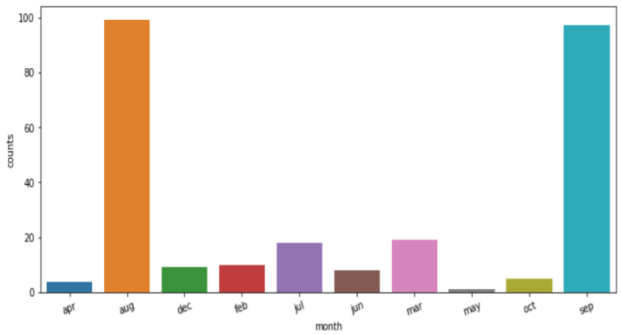

Fig.20 Bar graph showing the frequency of occurrence of forest fires with respect to each month.

Fig.20 illustrates that the forest fire frequently happens in the month of August and September. They occur less frequently in the month of April and May. During the other months their occurrence is sporadic and not as much as they occur during August.

Retrieval Number: B2878078219/19©BEIESP
DOI: 10.35940/ijrte.B2878.078219
Journal Website: www.ijrte.org

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication

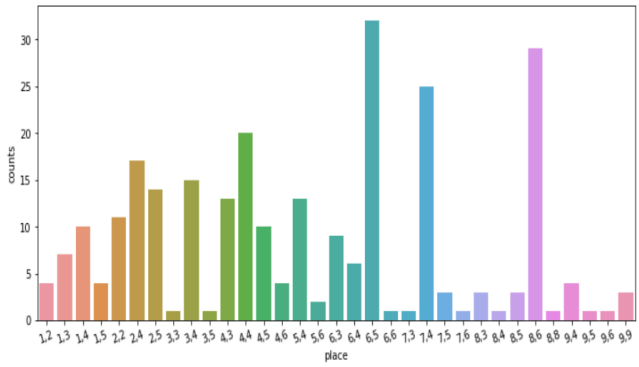

Fig.21 Bar graph showing the frequency of occurrence of forest fires with respect to each place.

The wildfires happen frequently in the place where the X-and Y- spatial coordinates within the park are 6 and 5, respectively. The places where they are less frequent are at (3,3);(3,5);(6,6);(7,3);(7,6);(8,3);(8,4);(8,8);(9,5);(9,6) as shown in the Fig.21.

	FFMC	DMC	DC	ISI	temp	RH	wind	rain	fire
count	517.000000	517.000000	517.000000	517.000000	517.000000	517.000000	517.000000	517.000000	517.000000
mean	90.644681	110.872340	547.940039	9.021663	18.889168	44.288201	4.017602	0.021663	0.522244
std	5.520111	64.046482	248.066192	4.559477	5.806625	16.317469	1.791653	0.295959	0.499989
min	18.700000	1.100000	7.900000	0.000000	2.200000	15.000000	0.400000	0.000000	0.000000
25%	90.200000	68.600000	437.700000	6.500000	15.500000	33.000000	2.700000	0.000000	0.000000
50%	91.600000	108.300000	664.200000	8.400000	19.300000	42.000000	4.000000	0.000000	1.000000
75%	92.900000	142.400000	713.900000	10.800000	22.800000	53.000000	4.900000	0.000000	1.000000
max	96.200000	291.300000	860.600000	56.100000	33.300000	100.000000	9.400000	6.400000	1.000000

Fig.22 Tabular representation of the description of each features.

The feature 'rain' has a standard deviation of 0.29 and a variance of 0.09 which is closer to 0 as shown in the Fig.22. Features that have variance closer to 0 do not have any effect on the model for prediction. So, therefore the features that are considered for modeling are FFMC, DMC, DC, ISI, temp, RH, wind.

The features FFMC, DDMC, DC, ISI, temperature, RH, wind are encoded for categorization. The above process is done to encode similar decimal values with the same value. This ensures that the scaling is done efficiently. The Min-Max scaling is done after the values of the dataset are encoded. Min-Max Scaling converts all the values of the dataset within the range 0 to 1.

III. NORMALIZATION TECHNIQUES

A. Principal Component Analysis (PCA)

When the number of features in the dataset is large, and there exists correlation among certain features, then PCA is used to convert the higher dimensional correlated features into lower dimensional uncorrelated features [12], [13]. Thus PCA removes redundancy in the dataset, and also achieves a reduction in dimensionality. The new set of variables or components are orthogonal to each other with less dependency. The steps involved in performing PCA are as follows:-

(i) Computing the mean and covariance of the matrix (data set)

$$S = \frac{1}{n}\sum_{i=1}^{n}(\vec{x}_i - \vec{x})(\vec{x}_i - \vec{x})^T \quad (1)$$

Where,

S is the covariance of matrix $'x'$

$$\vec{x} = \frac{1}{n}\sum_{i=1}^{n}\vec{x}_i \quad (2)$$

The above equation is to find the mean of the variable.

(ii) Performing Singular Vector Decomposition (SVD)

$$S = U\Sigma V^T \quad (3)$$

where,

$$U \in \mathbb{R}^{n\,x\,n}, \Sigma \in \mathbb{R}^{n\,x\,n}, V \in \mathbb{R}^{n\,x\,n}$$

(iii) Projecting the target matrix,

$$Y = P^T X \quad (4)$$

B. Feature Scaling

Feature scaling is a method used to standardize the range of variables of a data [14]. It is basically normalizing the data and generally done during the pre processing step. This paper uses Min-Max scaling for normalizing the data.

1) Min-Max Scaling

It is a scaling where the range of values is scaled within the range of [0,1] or [-1,1]. The target range is selected based on the nature of the data [15]. The relationship among the original data is preserved after the normalization.

$$x' = \frac{x - \min(x)}{\max(x) - \min(x)} \quad (5)$$

where, x is the value to be normalized

$\min(x)$ is the minimum value that $'x'$ takes in the column.

$\max(x)$ is the maximum value that $'x'$ takes in the column.

C. Label Encoding

Label encoding enumerates each and every value in a column and also similar values are given similar numbers. The categories 'yes' and 'no' are encoded as 1 and 0. Since, algorithms like XGBoost do not support categorical values and only numerical values, those values should be enumerated. It is similar to Ordinal encoding where the features other than the target variable are encoded and in Label encoding the target variable is also encoded additionally[16].

IV. MACHINE LEARNING TECHNIQUES

Some of the techniques used for solving regression and classification problems used in this study are discussed as follows.

A. Logistic Regression

Logistic Regression is a machine learning method used for modeling a binary dependent variable [17], [18], [19]. It is a form of binomial regression. The dependent variable takes a binary form – 1 or 0, yes or no. The relationship between

the dependent variable and the independent variable helps it to predict the target variable. Logistic regression uses sigmoid function to determine their probability and map them to some discrete values. The sigmoid function is as follows:-

$$\phi(z) = \frac{1}{1+e^{-z}} \quad (6)$$

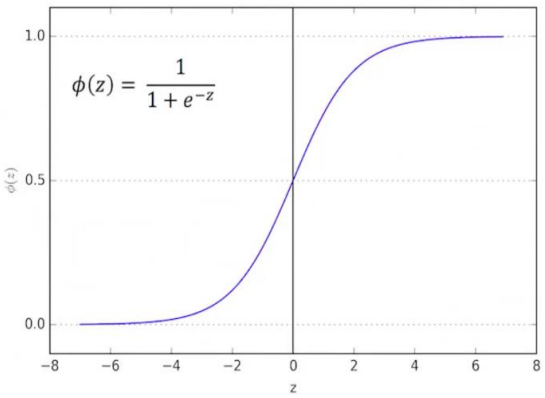

Fig.23 Graphical representation of sigmoid function

Fig.23 illustrates that the value of $\phi(z)$ is within the range of [0,1]. Hence it can be said that $\phi(z)$ represents the probability of the occurrence of 'z'.

B. Support Vector Machine (SVM)

The data is plotted in N-Dimensional space where the coordinates in the plot corresponds to its value [20]. Then a hyper-plane is found which distinguishes the two classes as shown in the Fig.24. The points in the plot are called as support vectors.

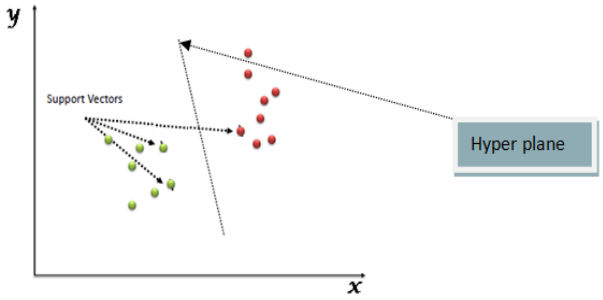

Fig.24 Graphical representation of hyper-plane and the support vectors.

The most important aspect of SVM's is to find the right hyper-plane [21], [22]. The right hyper-plane has the maximum width among all the plausible hyper-planes so that the number of data items that are classified incorrectly is minimized. The right hyper-plane also eliminates over-fitting and under-fitting of training data. The present work uses Radial Basis Function (RBF) as the kernel for SVM.

C. Random Forest

Random forest uses ensemble learning to construct multiple decision trees using the training set. During classification it computes the mode of the results obtained from the individual trees and during regression it computes the mean of the results obtained from the same [23]. The advantage of random forest over decision tree is that they correct the over-fitting nature of the decision trees. Random forest algorithms are very efficient in working with image data to classify them [24].

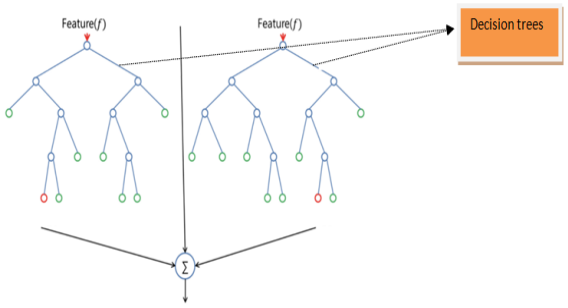

Fig.25 Overview of the two decision trees that constitute a Random forest.

From the Fig.25, two separate decision trees are taken and the outputs from both trees are taken and then summated to give a final decision on the classification of a data.

K-Nearest Neighbors (KNN)

KNN is an instance-based learning algorithm, where only a local approximation of the function is done [25], [26]. For the classification of a data, its 'k' nearest neighbors is collected and a neighborhood for the data is formed [27]. Then a voting is done locally amongst the neighborhood to classify the data. K-nearest neighbor algorithm is very sensitive to local data.

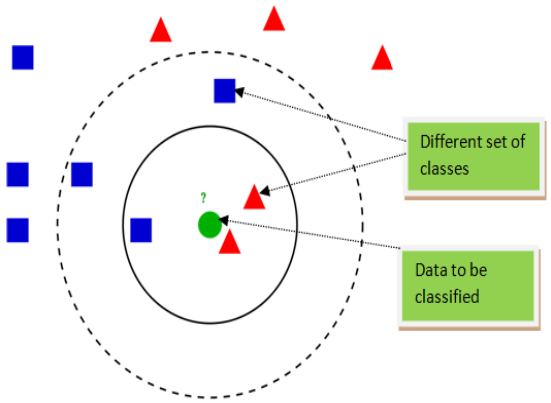

Fig.26 Representation of the 3 classes classified based on their closest neighborhood.

The green circle from the Fig.26 has to be classified correctly to its class. It has to be classified either to the 'blue' class or to the 'red' class. Since its neighborhood has 2 data belonging to the red class and only 1 belonging to the blue class, the green circle is classified as a data belonging to the red triangular class.

D. Bagging Classifier

Bagging is an ensemble machine learning method where ensemble is created by combining individual classifiers like decision trees. Each classifier's (tree) training set is generated by randomly drawing from the original set [7], [9]. In this work, the classifiers used are decision trees. Bagging predictor improves the accuracy of a model by applying a bootstrap sampling to the training dataset. Bagging is generally done to reduce a model's variance and to control over-fitting.

E. Boosting classifier

Boosting is an ensemble algorithm, which uses a set of classifiers, each having a low accuracy ratio (but not <50%), and hence considered 'weak' to build a 'strong' classifier [7], [8]. These classifier are trained in sequence, where the error obtained from the i^{th} model is used to improvise the prediction of $(i+1)^{th}$ model.

1) Gradient Boosting

Gradient boosting trains many models in an additive, sequential and measured manner [28]. This algorithm identifies the drawbacks of weak learners by using gradients in the loss function. The loss function is a measure that depicts how efficient are the model's coefficients at fitting the data [29]. In gradient boosting function the users are allowed to optimize the cost function instead of the loss function (which offers very less control).

2) AdaBoost Classifier

In AdaBoost algorithm, the individual classifiers and the data are assigned with weights such that the model's focus shifts to those data that are not classified correctly [30], [31]. The algorithm executes the process sequentially and ensures that the weights are correctly modified after each step. The final Hypothesis is given by,

$$H(x) = sign\left(\sum_{t=1}^{T} \alpha_t h_t(x)\right) \quad (7)$$

Initially, a weak learner is trained for a distribution 'D'. After that, the hypothesis takes a low weighted value such that the condition for ε_t is satisfied. Then accordingly the value the value of α_t is chosen. After each iteration, the value of $D_{t+i}(i)$ is updated and then the final hypothesis is formulated.

3) XGBoost

XGBoost (XGB) stands for "Extreme Gradient Boosting". It is called so because of its speed and efficient computation when compared to the other gradient boosting algorithms [32]. It has been proven to push the limits of computation power with efficiency to achieve greater results. It is similar to gradient boosting apart from the fact that it performs an additional custom regularization in the objective function.

V. RESULTS

Table.1 Tabulation of F-1 score with respect to each Machine learning algorithm after performing PCA.

Algorithm	*Accuracy*
Logistic Regression	**68.26**
Bagging Classifier	61.7
AdaBoost	57.92
Gradient Boosting	54.87
XGB	56.79
SVM	42.26
Random forest	51.14
KNN	51.14

The dataset was modeled in two different ways, one where PCA was applied and the other without PCA. For the first case, after applying PCA, forming a table with 5 principal components, the dataset was then split into training and test data, with 20 percent of the dataset being the test data. After that the ML techniques were applied for modeling and the results were tabulated (see Table.1). For Logistic Regression, it gave the highest F-1 score of value 68.26. Then it was modeled with Bagging classifier which gave a F-1 score of 61.70. The AdaBoost classifier gave a value of 57.92. The F-1 score for XGB was recorded at 56.97. The gradient boosting gave a value of 54.87. Both Random Forest and K-NN (with 5 neighbors) models gave an accuracy of 51.14. The Support Vector Machine, with RBF, gave the least value among the models that were applied with PCA. The value for SVM model was 42.66.

Table.2 Tabulation of F-1 score with respect to each Machine learning algorithm without performing PCA.

Algorithm	*Accuracy*
Logistic Regression	66.69
Bagging Classifier	53.69
AdaBoost	66.51
Gradient Boosting	**68.38**
XGB	63.62
SVM	66.33
Random forest	50.13
KNN	66.69

The second case was to model the dataset without applying Principal Component Analysis. Then the dataset is then split into the test and train data, with test data constituting 20 percent of the original dataset. After splitting, both the test and train data are scaled within the range of 0-1 using Min-Max scaling. Once the splitting was completed, the models were created and the results were tabulated see (Table.2). For Logistic Regression, the value for F-1 score was 66.69. Then it was modeled with Bagging classifier with a F-1 score of 53.69. The AdaBoost classifier gave a value of 66.61. The F-1 score for XGB was recorded at 63.62 .The value for Gradient Boosting was 68.38, the highest value among the models in the second case. The Random Forest model gave an accuracy of 50.13. The K-NN (with 5 neighbors) gave an accuracy of 66.69. The Support Vector Machine, with RBF kernel, gave a value of 66.33.

For the first case the logistic regression gave the highest value at 68.26 and SVM gave the lowest value at 42.66. In the second case, Gradient Boosting classifier gave the highest value at 68.38, which turned out to be the highest accuracy among all the models that were taken into consideration (both Case-1 and Case-2). The least value in the second case was given by Random forest classifier with a value of 50.13.

VI. CONCLUSION

In this paper, the forest fire dataset from the UCI machine learning repository was used to predict the occurrence of a wild fire. To predict its occurrence, a new target variable 'fire' was created from the feature 'area' to make the prediction a binary classification (whether a forest fire has occurred or not).

The results suggest that with PCA, Logistic regression's performance was the best among the models considered with an accuracy of 68.26. Without PCA, Gradient Boosting gave the best performance with an accuracy of 68.38 (Highest score with both the cases considered). The work can be extended further by using deep learning models such as Artificial Neural Networks (ANNs) and also by predicting the area of the forest burnt using various regression models.

REFERENCES

1. Soo Chin Liew, "Satellite detection of forest fires and burn scars", 2001
2. P. W. Adriaans, D. Zantinge, Data Mining, Addison-Wesley, 1996.
3. Tukey. J. W. (1962)," The future of data analysis", Ann. Statist.33 , 1-67
4. S. Agarwal, "Data Mining: Data Mining Concepts and Techniques," *2013 International Conference on Machine Intelligence and Research Advancement*, Katra, 2013, pp. 203-207.
5. P. Cortez and A. Morais, "A Data Mining Approach to Predict Forest Fires using Meteorological Data.", In J. Neves, M. F. Santos and J. Machado Eds., New Trends in Artificial Intelligence, Proceedings of the 13th EPIA 2007 - Portuguese Conference on Artificial Intelligence, December, Guimarães, Portugal, pp. 512-523, 2007, APPIA, ISBN-13 978-989-95618-0-9.
6. G. E. Sakr, I. H. Elhajj, G. Mitri and U. C. Wejinya, "Artificial intelligence for forest fire prediction," *2010 IEEE/ASME International Conference on Advanced Intelligent Mechatronics*, Montreal, ON, 2010, pp. 1311-1316.
7. Bühlmann, Peter, "Bagging, Boosting and Ensemble Methods", Handbook of Computational Statistics. 10.1007/978-3-642-21551-3_33, 2012
8. T. M. Khoshgoftaar, J. Van Hulse and A. Napolitano, "Comparing Boosting and Bagging Techniques With Noisy and Imbalanced Data," in *IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans*, vol. 41, no. 3, pp. 552-568, May 2011.
9. L. Breiman, "Bagging predictors", *Machine Learning*, vol. 24, no. 2, pp. 123-140, 1996.
10. Goutte, Cyril & Gaussier, Eric, "A Probabilistic Interpretation of Precision, Recall and F-Score, with Implication for Evaluation", Lecture Notes in Computer Science. 3408. 345-359. 10.1007/978-3-540-31865-1_25, 2012
11. D. Zhang, J. Wang, X. Zhao and X. Wang, "A Bayesian Hierarchical Model for Comparing Average F1 Scores," *2015 IEEE International Conference on Data Mining*, Atlantic City, NJ, 2015, pp. 589-598. doi: 10.1109/ICDM.2015.44
12. Chandra Paul, Liton & Suman, Abdulla & Sultan, Nahid, "Methodological analysis of principal component analysis (PCA) method", International Journal of Computational Engineering & Management, 16. 32-38, 2016
13. Jonathon Shlens," A Tutorial on Principal Component Analysis", Educational, 51, 2014
14. Bikesh KumarSingh , Kesari Verma & A.S. Thoke, "Investigations on Impact of Feature Normalization Techniques on Classifier's Performance in Breast Tumor Classification", International Journal of Computer Applications. 116. 11-15. 10.5120/20443-2793, 2015
15. S.Gopal Krishna Patro & Kishore Kumar Sahu, "Normalization: A Preprocessing Stage", IARJSET, 10.17148/IARJSET.2015.2305, 2015
16. Kedar Potdar, Taher Pardawala, & Chinmay Pai, "A Comparative Study of Categorical Variable Encoding Techniques for Neural Network Classifiers", International Journal of Computer Applications. 175. 7-9. 10.5120/ijca2017915495, 2017
17. Joanne Peng, Kuk Lida Lee, & Gary M. Ingersoll, "An Introduction to Logistic Regression Analysis and Reporting", Journal of Educational Research - J EDUC RES. 96, 3-14. 10.1080/00220670209598786, 2002
18. T. Haifley, "Linear logistic regression: an introduction," *IEEE International Integrated Reliability Workshop Final Report, 2002.*, Lake Tahoe, CA, USA, 2002, pp. 184-187.
19. D. W. Hosmer, S. Lerneshow, "Applied Logistic Regression", Wiley Interscience, 2000.
20. M. A. Hearst, S. T. Dumais, E. Osuna, J. Platt and B. Scholkopf, "Support vector machines," in *IEEE Intelligent Systems and their Applications*, vol. 13, no. 4, pp. 18-28, July-Aug. 1998.
21. Yujun Yang, Jianping Li and Yimei Yang, "The research of the fast SVM classifier method," *2015 12th International Computer Conference on Wavelet Active Media Technology and Information Processing (ICCWAMTIP)*, Chengdu, 2015, pp. 121-124.
22. Kwang In Kim, Keechul Jung, Se Hyun Park and Hang Joon Kim, "Support vector machines for texture classification," in *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 11, pp. 1542-1550, Nov. 2002.
23. L. Breiman, "Random forests", *Machine learning*, vol. 45, no. 1, pp. 5-32, 2001.
24. T. K. Ho, "Random decision forests", *Proceedings of the Third InternationalConference on Document Analysis and Recognition*, pp. 278-282, 1995.
25. R. Agrawal, "K-Nearest Neighbors for Uncertain Data", *International Journal of Computer Applications (0975–8887)*, vol. 105, no. 11, pp. 13-16, 2014.
26. Okfalisa, I. Gazalba, Mustakim and N. G. I. Reza, "Comparative analysis of k-nearest neighbor and modified k-nearest neighbor algorithm for data classification," *2017 2nd International conferences on Information Technology, Information Systems and Electrical Engineering (ICITISEE)*, Yogyakarta, 2017, pp. 294-298.
27. X. Wu et al., "Top 10 algorithms in data mining", *Knowl. Inf. Syst.*, vol. 14, no. 1, pp. 1-37, 2008.
28. T. G. Dietterich, G. Hao, A. Ashenfelter, "*Gradient Tree Boosting for Training Conditional Random Fields".*
29. J. H. Friedman, "Greedy function approximation: A gradient boosting machine", *Annals of Statistics*, 2000.
30. Ying Cao, Qiguang Miao, "Jiachen Liu. Adaboost algorithm research progress and prospect", *Journal of automation*, vol. 39, no. 6, pp. 745-758, 2013.
31. X. Shu and P. Wang, "An Improved Adaboost Algorithm Based on Uncertain Functions," *2015 International Conference on Industrial Informatics - Computing Technology, Intelligent Technology, Industrial Information Integration*, Wuhan, 2015, pp. 136-139.
32. Tianqi Chen & Carlos Guestrin, "XGBoost: A Scalable Tree Boosting System". 785-794. 10.1145/2939672.2939785, 2016

AUTHORS PROFILE

Rishickesh Ramesh is working toward B.Tech degree in the Department of Information Technology, SSN College of Engineering. His research interests include machine learning techniques for natural language processing and understanding, data analytics, Internet of Things and deep learning.

A. Shahina is a professor in the department of Information Technology at SSN. She has 14 years of teaching and research experience, with over 5 years of research exclusively in the field of Speech Processing, one of the widely growing and popular research areas. She aims to develop speech based clinical applications, and technologies for viable biometric person authentication systems through sustained research. Her areas of interest include machine learning, deep learning, and speech processing.

Dr. A. Nayeemulla Khan is the Dean Academics and Professor in the School of Computing Science and Engineering at VIT Chennai. He was previously associated with the Airports Authority of India as a Senior Manager ATC and as a Research Scientist at Acusis Software India Pvt. Ltd. His areas of interest include speech and speaker recognition, machine learning, brain computer interface among others.